MICRO COMPUTER

PRINTOUT

A PLAIN MAN'S GUIDE TO PERSONAL COMPUTING

September 1982
95p

Word Processing
What to buy

GUARANTEED COMPUTERS

'S NEW & USED

VERY RELIABLE

ONE LADY OWNER

GOOD VALUE SNIP!

LOVELY PERFORMER

STAR BUY

BARGAIN MODEL

FREE!

How to choose your dealer

ORNE 1

™

Internal electronics. Z80A™ CPU, 64K bytes RAM memory (60K available to the programmer, 4K used to run the screen.) System software is held in ROM in a separate address space.

Screen. Clear, 5", 24-row screen displays a 52-character window on a 128-character line with automatic horizontal scrolling.

Monitor Interface. Connects the OSBORNE 1 to any monitor screen.

Keyboard. A standard typewriter keyboard plus numeric, adding-machine keypad for fast entry, and cursor control keys for easy cursor movement.

Case. The plastic case snaps together to form a weatherproof, 24-pound package that fits underneath the standard airline seat.

Standard software
Five outstanding software packages, with a retail value of over £800 are included:
● CP/M® Operating System
● WORDSTAR® with MAIL MERGE®
● SUPERCALC™ ● MBASIC®
● CBASIC®

Optional extras
● Modem cable for use with acoustic couplers for telephone transmission of data
● Battery pack
● Double density disk drives with 200K bytes of storage per drive

Trademarks: SUPERCALC: Sorcim Corporation; Z80A: Zilog Corporation.
Registered Trademarks: OSBORNE 1: Osborne Computer Corporation; CP/M: Digital Research; WORDSTAR, MAIL MERGE: MicroPro International; MBASIC: Microsoft; CBASIC: Compiler Systems, Inc.

mputer you can take anywhere.

For further information and full specification, return the coupon to The Marketing Manager, Osborne Computer Corporation (UK) Ltd. 38 Tanners Drive, Blakelands North, Milton Keynes, Buckinghamshire MK14 5BW. Telephone: 0908 615274. Telex 825220

More information on Osborne 1, please.

Name

Address

Tel

MCP/9

OSBORNE
COMPUTER CORPORATION (UK) LTD.

SMALL FIRE IN SCOTLAND NOBODY HURT.

The world is divided into those who have Petspeed and those that don't. A third sub-group – those without Commodore – is too sad to dwell on.

But why should the obvious excite the passions, or any interest at all? 'Have' and 'have nots' have been around since Adam snatched the apple (sorry). And look where that got him, you might say.

We make no apology for being evangelistic about Petspeed. It has changed the lives of hundreds of Commodore users and we are dedicated to the idea that they should not remain an elite.

We also believe that this article should not be as boring as the headline (a classic in its time.)

So we are not going to list all the benefits and advantages of faster compiling. Or tell you how the magic of Petspeed works. You can find out by contacting us for a demo disc.

But we do have one bit of news (if you haven't already heard) Commodore UK have commissioned Oxford Computer Systems to produce an advanced Petspeed.

Now that's news.

to discover the meaning of life and report back in a million years.)

Mark is the OCS equivalent of college chaplain though he claims he's never had a job title. He stopped shaving when he was fourteen – when presumably he discovered other things – and has been contemplating ever since.

He came to Oxford from Santa Clara via Slough. "I missed England when I was in the States and now I miss the States." In his new role with the Oxford hardware design team he will be happy – spending part of the year on each side of the Atlantic.

Son of Wilkes qualifies. Again.

Not content with teaching music, Tony Wilkes son of the celebrated Cambridge professor Maurice Wilkes, returned to his studies and finished up with a degree in Electrical Engineering which he collected from Manchester University last month.

In his spare time during his full-time degree course Tony designed the PETSPEED Compiler.

Now, back among the dreaming spires, he and his partner Alan Wyn Jones are working full time on a series of new products for the micro market.

Double Trouble

Dave Haines, an 'ace database guy' from the Rutherford laboratories, and Mark 'Guru' Clarke from Commodore's West Coast hardware/software design team have recently added new blood, zest, enigma and brainpower to Oxford Computer Systems.

Dave left St John's College with a degree in psychology and the honour of being the first 'space wanderer' elected by the junior common room. (Duties:

GENUINE PETSPEED OFFER FOR READERS OF MCP.

Oxford Computer Systems are giving away a year's subscription to this magazine plus a free copy of THE PET COMPANION with every PETSPEED Compiler sold during September using the coupon below.

The PET COMPANION is a compilation of all the PET material published in the first volume of MCP. It includes over a hundred PET programming hints and tips in addition to 27 major articles on the subject.

Cut and complete the coupon to qualify.

Laid-back Larry Competition

Our silent friend Laid-back Larry was the unheard star of the latest Commodore Exhibition at the Cunard Hotel.

Larry was bumped into and nudged several times during those three hectic days. He never complained.

Most times he got an apology. All you have to do is guess how many times.

First prize is a jar of Cooper's Oxford Marmalade. Second prize, a weekend in Cambridge. Third prize, two weeks in Cambridge.

(Closing date for entries 1st August 1982. Judges decision is final.)

IF THE CLOUDHOPPERS, WASPS, SPIDERS, METEORS, OR SATELLITES DON'T BEAT YOU...

...RENAISSANCE WILL

CLOUDBURST
meteor run
Renaissance
SPIDERS OF MARS
satellites
and meteorites

5 More great
VIC Cartridge games from Audiogenic

What is a computer anyway?

Information Technology '82 has done a great deal to convince ordinary people that they need to know about computers. Yet few people can actually define what a computer *is*. In the first of our series of explanations for absolute beginners, **Richard Pawson** compares the computer with a food mixer.

What is a computer, anyway?

A computer, like any other kind of machine, is really a *processor* – that is to say, you feed something in, the machine performs a process on it and gives you something else out. Let's look at the example of a domestic food mixer. The most sophisticated models on the market which can chop, grate, slice, as well as mix, are even called Food Processors. Were you to set the controls correctly, you could feed in some carrots, white cabbage, onion and mayonnaise (let's call this our *input*) and the food processor gives you back ready-made coleslaw – the *output*.

Though some computers appear to deal with physical quantities (such as the computers which control huge coal fired power stations, or the robots which built Japanese cars), what all computers are really processing is *information*, but with many different ways of gathering the input data and issuing the output data.

In the power station, for example, the computer gathers its information about furnace temperature, power production and cooling water flow from a number of sensors in just the same way as the eyes, nose and ears feed input information to the brain. Some calculations are performed and the resulting data fed to a valve on the fuel hopper which controls the rate at which coal is fed in.

An example of a computer processing *pure* information is the typical desktop business micro. Information is fed in usually via the keyboard, perhaps in the form of sales orders, or figures from clock cards. This same information is then processed, combined with data that has been typed in previously and stored, and output in a different form – say invoices or Pay Slips.

Fast and reliable

Great, I can hear you saying, so why use a computer just to change figures from one form to another? Well, sometimes the advantage is speed. Engineers have to perform calculations that involve perhaps a million multiplications and three million additions – which even on a sophisticated calculator could take months if not years. Computers are also more reliable and can work continuously without breaks, which is obviously essential in our power station.

In most situations the same raw data is used to produce different results. In a typical business, information from sales orders can be processed to tell the Marketing Manager which regions are selling best, the Credit Control department how much money is owed, and the warehouse which product lines need re-ordering. By using a computer once an item of information has been typed in, it is *stored* or memorised and can be used in future calculations without the need for retyping.

But the area in which computers really score over their human counterparts is in repetitive processes or calculations. If you start to think about it, an awful lot of the

processes we perform are repetitive. Sometimes simple like adding VAT to an invoice, sometimes complex like extracting a statistical trend from some scientific data. Once a computer has been shown how to perform a process, it can repeat the technique ad infinitum – using different information as input, and hence different output each time.

Telling the computer how to perform a process the first time is called *programming*. Programming is needed because computers aren't really very clever and can only do straightforward things like adding two numbers together. Programming simply involves breaking down your *application* into successively smaller tasks or operations until each one can be understood and performed by the computer.

Set of instructions

The resulting list of operations or simple tasks is consequently called the *program*. In this respect a computer program is no different from a recipe, knitting pattern, or set of instructions for assembling a kitchen cabinet. Like all the above, a computer program is written in a special kind of language or jargon consisting of many abbreviations, strange symbols and phrases. Which makes it all the more surprising that people who can fluently read the complex coding of a knitting pattern believe they can't get to grips with computer programming.

The language in which programs are written (there are several languages e g FORTRAN, COBOL and PASCAL though BASIC is the most commonly used in micros) has been designed to assist you, the user. First it saves space by getting you to define operations in a concise form. Writing

TOTAL = PARTS + LABOUR

is obviously quicker than saying

Add the values of PARTS and LABOUR and then place the result in TOTAL

Secondly, the programming language encourages you to be *precise* defining what you want to do in a manner that can't be interpreted ambiguously.
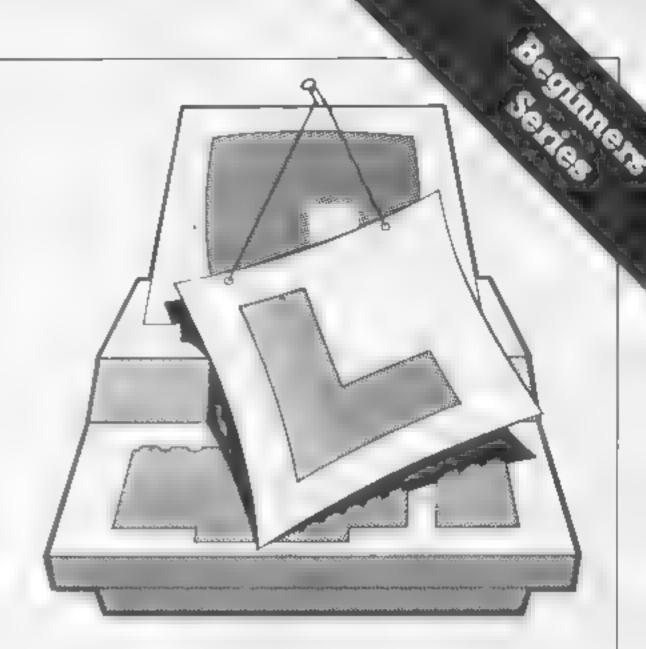
The main function of a programming language is to provide you with tools for performing the most common tasks. It has already been stated that the Central Processing Unit of a computer is very dumb, most can't even perform a multiplication function. The programming language provides you with such a function (plus many more advanced features) and every time this is invoked, the language translates the multiplication into hundreds of simple additions which the computer understands, though you don't see any of this happening.

Writing the program in a more sophisticated language (the higher the level of a language, the closer it is to plain English), not only saves time, but makes the list of instructions easier to read, and helps avoid mistakes by eliminating the need for hundreds of statements to perform even a simple task.

Summary

So what have we learnt in the first instalment? Here's a quick summary -

1 A computer processes information. It takes in data in one form (INPUT) performs calculations on that data and feeds it out in another form (OUTPUT)

2 The list of instructions that tell the computer how to perform a particular process is called a program. Once

programmed, a computer can repeat the same process on different data very quickly

3 Programming is simply the technique of breaking down your application into small chunks which the computer can understand. These individual instructions are expressed in a programming language which provides tools for all the most common tasks. This language is translated automatically by the computer into a very long list of ultra-simple commands which its Central Processing Unit can follow.

Next month we shall be looking at the different parts that make up a typical computer system, and what role each plays in processing information.

"... all computers are really processing information, but with many different ways of gathering the input data and issuing the output data"

READ/WRITE

Big deal

I note that several printer retailers are offering quite large price reductions on printers if more than one are bought. Discom, for instance, offer the Smith-Corona TP-1 Text printer at £485 for one; £395 for two; £340 for 50

Wouldn't it be a good idea if some micro users got together and formed a kind of bulk buy co-operative? I'd be happy to join one

T M Artingstoll
Zwoller, The Netherlands

Excellent idea! That is precisely how some dealers started life. MicroComputer Printout wholeheartedly supports the idea of a computer-users co-operative — and we would certainly give publicity to anyone willing to start one

De Re Atari

Further to my earlier letter concerning the whereabouts of the Atari manual 'De Re Atari' I am able to give a U K. Distributor's address

Details of nearest retailers are available from

Atari U K
WEA Records.
London W1V 2BH

or contact

Maplin Electronic Supplies.
P O Box 3.
Rayleigh, Essex SS6 8LR

who have some copies of the manual

If you could print these addresses as an update of the earlier U S. address I sent. I am sure they will be well received

N. Crozier,
Malton, N Yorkshire

Rogues Gallery

Your contributors range from the sublime (Tommy, Terry Hope) to the ridiculous (Richard Nicois, Inside Trader) What about brief biographies of this bizarre crew?

Harvey Smith
(not '2 fingers' Harvey)
Halifax

Unfortunately, this letter went right round the office before the idea could be suitably squashed. Even before the Editor could get started on Volume Thirteen of his memoires, his desk was knee deep in submitted articles along the lines

'A day in the life of Lindsay Doyle
'Terry Hope — the early years
Mein Kampf' — Tommy
'Pubs I have known' — Martin 'Legless' Banks

Expect some suitably silly excerpts in forthcoming issues.

Male Chauvinism (part I)
Re "User Friendliness"

I should like to express my distate with re-gard to the above-mentioned article by Chris Preston, appearing in your August 1982 issue. I take exception not to Mr Preston's points concerning the evaluation of user friendliness in micro computers, but rather to the manner in which these points were made His crude generalisations regarding the nature of female computer operators (whom he characterises either as mindless school leavers or "slightly dotty old matrons, from 40 up to 60+"), the level of their skills and commitment to their jobs, were both offensive and unnecessary in the context of the subject of the article Surely it is possible to reserach, write and edit an article such as this without denigrating women and the role they play in operating business computers. I strongly suggest that if Mr Preston is not capable of so doing, and his work makes it glaringly obvious that he is not, then a woman author might have been a better choice for this assignment

Kelsey Sterling
Oadby Leics

Surely you don't think that this magazine's initials are M C P by coincidence, Ms Sterling?

We're sure that Chris intended no offence in his comments about female computer operators — indeed, he is extremely fond of several. But when you have spent as many years as he has designing "fool proof" programs and watching people make seemingly impossible blunders, you are bound to end up with a rather cynical outlook

We would positively invite more contributions from our fairer-sexed readers (that's women you oafs!). They should write in confidence to the Editor, marking the envelope personal and confidential!

Male chauvinism (part II)

Last month's headlines in the Hotline news section ('Net Stockings', 'Debugger Off] were a blatent example of male chauvenism

elitist pornography and rampant sexism facile fascist opportunist rubbish ... press bias ... degrading women and gays ... exploitation of women ... Your columnist should be castrated

Grinda Glotz
Women against Sex and Violence in the Media
Stoke Newington

This letter has been edited to remove the sex and violence

Cheap Monitor

With reference to S Macenzie's letter in your August issue yes is is possible to convert a portable TV set to monitor use, further the conversion is cheap and easy However the following rules must be obeyed

1 The TV must have an isolated chassis
2 The mains lead must be three core with the chassis permanantly earthed (i e replace two core mains leads)

Having done that all that is necessary is to wire a resistor to turn off the fine. Vision IF transistor, and to feed the video signal into the base of the first video amplifier, ensuring that signals to the sync separator remain intact Bandwith should be OK for a ZX81 but could be increased by increasing stages and decreasing stage gain in the video amplifier

If none of this makes any sense take the set and this letter to your friendly local TV repairer

I have done this myself to a cheap Ingersol

READ/WRITE

TV and now have a £45 monitor which will re-solve 80X32 text from a BBC micro

G. Cox.

How many times must we warn readers not to contradict the advice given out by our boffins – they can act in a most unpredictable manner! (Mr Cox's address has been withheld in the interests of his continued good health)

Bugs squashed

I have just read the letter from G. Mayer in the July issue of *MicroComputer Printout* BEEBUG is currently investigating his claims about appointed dealers, but I can confirm that some are indeed taking such an attitude We will report in future issues of our newslet-ter

Secondly, in your reply at the end of the let-ter you state that owners of BBC Micros should join one of the 'Independent' user groups such as BEEBUG or Laserbug May I state categorically that as far as I know the only truly independent user group is BEEBUG Laserbug are run by a shop called 'Computers for All', and cannot represent their members in a truly independent way. For example they recommend to their mem-bers that to convert BBC machines from 16K to 32K that they buy the chips from them at over *twice* the price that BEEBUG has negotiated, thus costing their trusting mem-bers an extra £20 or so!

Sheridan Williams.
St Albans, Herts

We invite Laserbug to reply, defending their case

Ex-editor gets fuzzy

D600 E360
1000 r300 w300 i536 t000 a632 b000 B000
C140 o500 F200 M325 S340, i300 c535
s000 c000-u100 T000 v614 i500 f500 o500
h536 a530 e230 s430 b000 s000 c530 o100
t000 n160 o100 e453 t000 b000 s630 A536
c000-u100 c500 b000 f530 i500 t000 t230 –
t000 m260 e453 o100 t000 n000 a600 i200
t000 n000, t000 s530 e453 i200 t000 S532
c300

    K530 r263
    David
    D216 D100 T130

P000, S000, t000 o540 s300 t000 S400-
M325 p630 o100 t000 p626

We have always maintained that PCW was difficult to understand, now we know why – their ex-editor (D D T himself) has difficulty in getting to grips with plain ordinary English!

For those not blessed with an IQ of 280 (or Bob Chappell's Fuzzy Matching program), D D T is pointing out some errors in the arti-cle and listing Corrections are as follows

Line 180: N and M should be set to the value of TN e 130

Text: N$() First minor element holds the origi-nal name, second minor element holds the soundex coded name

Bob advises us that these do not affect the running of the program and are merely skeletons' left over from the development stage

A leg to stand on

Our attention has been drawn to references in your issues dated April, May, June, July and August 1982 to Martin 'Legless' Banks

We act for Mr Banks who states that he is neither an amputee nor an inebriate To suggest otherwise is to cast a most damag-ing slur on the reputation of a distinguished journalist

Unless an immediate retraction and a full apology is published in your next issue and appropriate compensation paid, we shall have no alternative but to advise Mr Banks of the legal remedies open to him

    Yours faithfully
    Slocumbe & Payne.
    Solicitors

We are happy to state that Mr Banks is one of the most sober journalists we know, we have sent him a bottle of Scotch by way of recom-pense

Quality of life

Congratulations on publishing a machine code program for LIFE on the ZX81 This is the first program for the ZX81 where you could set up the initial pattern that I have seen and gave much better results than the prog-rams with 'random' start patterns I've had LIFE running on my school's RML 380Z for a number of months now and found the follow-ing patterns give very worthwhile results on the ZX81 (Once the correct BASIC routine had been worked out and added to the pub-lished listing)

Firstly a simple line of five '0's quickly be-comes an alternating pattern called 'Traffic Lights', with a "life cycle" of two. That is, only two patterns are formed before they repeat Seven '0's in a line becomes a stable pattern of hexagons called 'Honey Farm' Ten '0's forms the 'Pentadecathalon' with a life cycle of fifteen

A more complex pattern is called 'Tum-bler', formed from this initial pattern

```
  00    00
  0 0  0 0
  0 0  0 0
   0    0
  00    00
  00    00
```

This pattern inverts itself every seven genera-tions and thus has a life cycle of fourteen

One of the simplest looking, but really very complex patterns, is called 'R-pentomino'

This is the starting pattern:

```
.       00
        00
        0
```

Given sufficient room to expand this pattern lasts 1102 generations The ZX81's screen is not large enough, however, to support the largest pattern generated so the pattern does not evolve to the full 1102 generations, though several gliders' are fired out as the pattern evolves

Another pattern, simply called Pattern, becomes stable after 173 generations This is its starting pattern

```
      000
      0 0
      0 0
```

Finally, two simpler patterns. This first is cal-led 'Spaceship and this moves horizontally across the screen

```
      0  0
      0
      0   0
      0000
```

The last pattern is three five dot, dashes This gives a very attractive display but again eventually overruns the ZX81's screen

```
.    00000 00000 00000
```

The problem of screen size is one I hope to solve when my BBC micro eventually arrives. *Ed – See Mr Steen's letter below for explanation of the word 'eventually'*

    Calum Steen
    Helensburgh

Soap opera

Now that the 'What IS happening at Atari?', 'Now it can be told' saga is over may I suggest that 'Hotline turns its attention to the problems with the BBC micro. After having waited for ten weeks to be told not to expect delivery until mid-August, I feel that Acorn/ BBC are running the kind of operation that would make even Clive Sinclair blush The microcomputer market in Britain may be only five years old but I would have thought that it was past the stage where a subcontractor to a major organisation can go into receiver-ship, as seems to have happened to Clear-tone

One is very much over a barrel in this situa-tion and it appears that the suppliers are quite happy to exploit this To cancel the order means losing the computer entirely and why should the makers care when there are plenty of other eager customers

Until the rush by the makers of these new micros to announce better and cheaper models is over, it seems that it is the buyer who must think twice before turning away from the tried and trusted models If I had the money than I wouldn't hesitate to buy an Apple II rather than finance a system that

Curse of the personalised mailshot

'Dear Mr Allason I am writing to you personally

Oh yeah? Sifting the genuinely personal letter from the personalised mass mailshot is becoming more difficult all the time

A year or so back mass mail shots were detectable by the sheer awfulness of the type quality either smudgy laser printing as favoured by Reader's Digest, or the dot matrix printing employed by less well heeled outfits

Since daisywheel prices crashed from around £2 500 to the current £500 mark detection has become much more difficult Microhacks who receive more of these letters than most, have learned to look for that evenness of typing and the limited character font that typifies the daisywheel

Now even these limitations are being swept aside in the drive to

achieve authentically manual i.e. bad typing

Zygal Dynamics (tel 08692 3361) are about to push out a _Diablo_ daisywheel printer that can run a wheel containing 192 complete characters and part characters, which can construct over 400 different symbols That compares with the 88 or 96 character wheels in current use
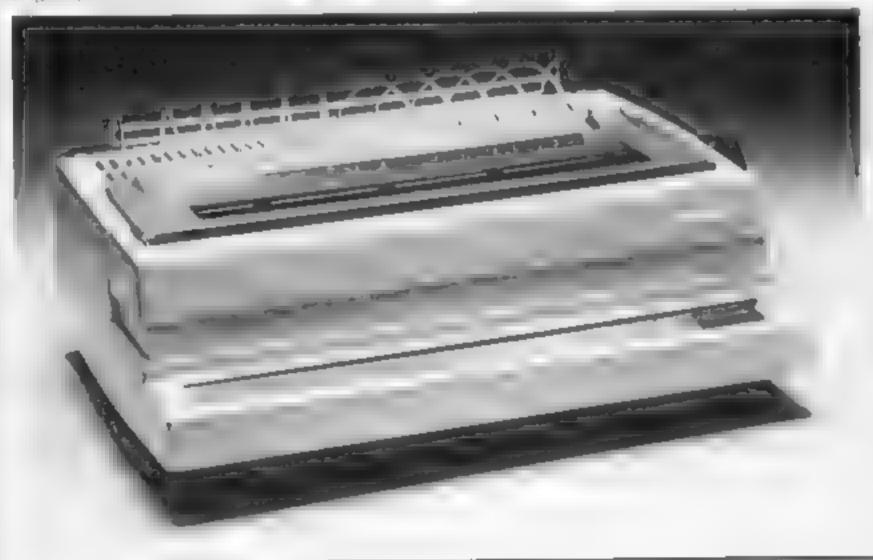
This new model designated the _630 ECS_ will be able to print standard and italic characters without changing daisywheels. It should be well received in technical and mathematical circles where daisywheel printers have long been cursed for their lack of special characters The mass mailers will probably buy it in droves

One of the new wheels will produce the complete international information

transmission set That means hard copy of Teletext screens. The ability to reproduce Teletext graphics may well also put further damper on dot matrix sales. Teletext graphics may not be quite up to hi-res reproduction of your lady wife but the character set is adequate for most types of business graphics, and even low resolution plotting

If you are considering electronic mail this could be the printer for you since the Teletext character set is the most likely standard to be adopted

Mind you, even electronic post offers you no escape from junk mail More than a third of the messages handled by one such Stateside utility, called the Source turned out to fall into this category And guess who owns it?

Readers Digest'

Micro disk madness

You want the good news or the bad news?

Right The good news is that four top floppy disk drive manufacturers – Dysan, Shugart Tabor and Verbatim have got around a table, and yes, they have agreed a standard For the new micro diskettes

These are the tiny three to four inch media that will provide the storage on the forthcoming generation of portable and pocket microcomputers

Standards yes, I know this column is obsessed with them covering 14" hard disks, 8 floppies, and, most familiar of all 5¼" minifloppies – have existed for some while now, and are almost universally accepted

Now comes the bad news

Standardisation of sizes never extended to read/write disk formats. In other words, a disk

made on an Apple will read back garbage if you try it in a SuperBrain And that goes for many supposedly compatible CP M diskettes too

I can't think of a single good reason why the same chaos won't ensue with micro diskettes Can you?

P S To further depress you here's news of a jumbo deal that Hewlett-Packard have just signed to use Sony's new 3½" disk

And guess what?

It is not compatible with the new 'standard'

S-100 standard not a standard – official

Never make a defence or apology before you are accused advised King Charles I before setting out for lunch with his bank manager

Sound enough advice, unless you happen to write about computers or the stock market for a living

So in anticipation of the accusation that this column ignores S-100 bus computers, I present the case for the defence

First a bit of background m'Lud

Early microcomputers were configured around a group of 100 lines, or bus, designated S-100 The theory was that additional printed circuit boards – interfaces graphics cards and the like – could be plugged into any computer adhering to the S 100 bus standard, and work

That at least was the theory

Such is the way of the world that different manufacturers soon discovered different ways of improving the S-100 standard In no time there were almost as many S-100 standards as manufacturers It certainly made plugging in an S-100 standard peripheral exciting

Some of the best microcomputers you can buy _Comart Communicator Cromemco_ and _North Star Horizon_ for example – use the S 100 bus 'standard' And their marketing people are much given to talking about the advantages of standardisation

Their marketing people are talking codswallop

Who says so? Not me (the Editor not to mention our libel insurers, wouldn't allow me to)

The microprocessor standards committee of the Institute of Electrical and Electronic Engineers does albeit in a more roundabout way

In fact they have just refused to classify S-100 as a standard, on the grounds that it suffers from a proliferation of options or "levels of compliance" as they call it

So although we will continue to report the wonderous advances made by Messrs Cromemco and company, you won't be reading much about the benefits of standardisation

Case rests m Lud

by Julian Allason

Sirius soft interface

Interfaces, ugh! Don'tcha hate em? I mean they look ugly, and rarely seem to do quite what you want them to.

Take IEEE-488 for example It's the standard means of connecting scientific instruments to computers, and the very *name* conjures up visions of unwashed boffins with beards

IEEE-488 was also one of the main reasons the early PETs established a foothold in the scientific market Until that is the bearded ones discovered that Commodore's idea of an IEEE-488 interface wasn't quite the same as the IEEE's (Institute of Electrical and Electronic Engineers)

Flavour of the month amongst the boffins just now is the ACT Sirius 1 – they like its all soft design. But the Sirius lacks IEEE 488

The penny dropped on the Hagley Road early this year when ACT salesmen started reporting

cases of Sirius-phobia amongst the men in white coats – due to lack of instrument interface

At this juncture the Brummies did something rather clever they pointed Harry Broomhall at a Sirius

Now Harold Algernon Edwin Broomhall FRAS may not be a name that is instantly familiar But amongst the Interface Crowd he's a star, if not *the* star A boffin's boffin, in short

Harry sat down and wrote a *software package* that reconfigures the Sirius I's parallel port to the IEEE-488 standard

This software consists of a set of high level routines written in assembler (Harry *thinks* in assembler), which can be called from a high level language such as BASIC Total hardware requirement one conversion cable

The Sirius is configured as a controller and can command other

devices on the bus to talk or listen

[Hey, if you don't understand this, don't worry, you don't need to]

What will be of interest however, is that it allows PET files to be fitted directly onto the Sirius

Whether or not Commodore are overwhelmed by this news well, would you want your customers upgrading to a competitive system? I think we should raise a glass to Harry

After all he has just invented the invisible interface

In the best possible taste

Do you belch after meals? Make offensive remarks when ladies enter the room? Wear wellies in your lounge?

Ha! We thought as much

With yobs like you around, it's no wonder the huntin', shootin' tax avoidin' set will have nothing to do with micros

Well it has got to stop And if you will kindly remove your finger from your nostril for one moment, I will tell you how you too can become a tasteful micro person

It will set you back about £900, mind, but bearing in mind the possible social spinoffs ("No, after you, Ma'am"), it's a snip at the price

It is our pleasure then to present Lancashire Furniture's new 'Technology with Taste' regency-style antique-look desk.

For those whose lack of refinement approaches the terminal, Lancashire Furniture also offer a special free Tasteful Design Service Have your butler call them on 0254-676011 today

---

Electronic non-message

Drat! It's 6 20 I've missed the post And this vital communication must reach my bookmaker by 9 30 tomorrow who insists, in writing (they don't accept begging telephone calls)

What to do?

Clearly this is a heaven sent opportunity to try out British Telecom's new fangled Telemessage 50 words plus address cabled to your addressee's nearest sorting office and delivered with the next morning's post

"Sorry, Sir, the Telemessage service closes at 6 p m " said a sepulchral voice at the other end of the phone It didn't *sound* sorry

But I thought the Telemessage was supposed to replace the telegram? I ventured

It is, Sir" said the voice Would you like to send a telegram instead?

I sent a telegram and my carefully worked out fifty words plus address cost a shattering £10 50. No wonder the service pretends it's dead

The solution to this nonsense must be electronic mail And if you lived in America you could have your microcomputer route a message to any one of the half million or so subscribers to the Source or CompuServe micronetworks

There is an option for us Limeys It is called Prestel, and I think it works something like this Subscriber A (that's you) dials up Prestel, climbing your way down a seemingly endless 'tree' of 'menus until you are through to the message box service Assuming you know Subscriber B's box number you can leave a message in her box

Most messages are apparently Please telephone me

I am not surprised

Note Perhaps readers can tell us more about Electronic Mail, as the Prestel Press Office doesn't answer letters and hasn't got a mailbox I wonder why?

Irresponsible corner

You think 'Bomb Buenos Aires was in bad taste?

Just wait til you see *Choplifter!* from Broderbund Software

The scenario has the player piloting a helicopter based for some reason, at an American Post Office He is tasked to rescue 64 kidnapped UN delegates who are being graphically zapped in a number of imaginative ways

"The joy of this game comes in the amusing graphics and sound " say the distributors

Apple owners with £21 95 burning a hole in their pockets can appal and disgust right-thinking relations by contacting Pete & Pam Computers on 0706-227011

Genie out of a bottle

What an odd lot *Genie* users are, and to judge by some of the red hot missives that seared their way into my in-tray this month, not inexpert with a sizzling simile either
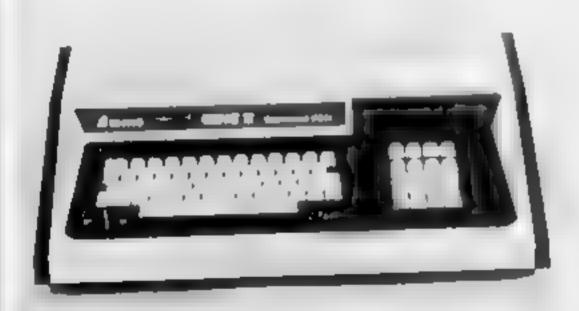
There was in fact an entirely valid reason why the *Genie* was omitted from last month's league table of topselling micros. It is this.

Some 11 000 odd Genie Is and IIs have indeed been sold here, according to our friends at Lowe Electronics. However, *current* sales are not what they were.

system always knows the time and date 640 x 288 pixel high resolution graphics will be available as an option. Ditto user definable characters

That's our scoopette. And you won't have read about the forthcoming *Colour Genie* anywhere else either

Like the VIC-20 it will allow up to 16 colours on screen at one time, it also offers 128 programmable graphic characters in addition to 128 that are preset 3 sound channels, full size typewriter keys and an internal

There is a good reason for this too. In fact, two good reasons

The first is the *Genie III*, whose appearance is imminent and whose resemblance to a certain Tandy Model III is entirely coincidental. Well, almost

You get 64K of RAM, 730K dual floppy disk drives and a 12" screen all built into a single unit It is Z80A based, of course, and CP/M and Level II BASIC is thrown in

There are several nice touches, including a CMOS real time clock with battery back up, so the

power supply. The normal screen format is 40 characters by 24 lines with 160 x 96 pixel resolution in graphics mode

At £199 incl. VAT the *Colour Genie* could give the VIC-20 and Atari 400 a run for their money, let alone the more expensive Tandy Colour system

None of this was supposed to be announced for a week or two yet but you know how it is with news – once it gets out of the bottle, you can't put it back

Just like a Genie

Machine Code for VIC

Nicely on top of BASIC are we? What about dipping a toe into assembly language then?

Hey, come back! It's not *that* difficult Especially if you have a well written assembler package at your disposal

VIC users have, thanks to those awfully nice Supersoft people *MikroAssembler* is described reasonably enough in our view – as an "advanced yet easy-to-use package which enables even novices to produce compact machine code programs." Well perhaps not complete novices

Anyway it is a cartridge which includes a Machine Code Monitor (the equivalent of the PET TIM) which facilitates the display of blocks of machine code for modification or storage on tape or disk As you would expect there is also an Assembler and facilities to preset the contents of the processor registers, insert and delete capabilities and a SAVE and VERIFY routine

Another nice touch is a high resolution plotting facility based on a 176 x 160 matrix, that lets you draw lines, plot points and even specify colour

£49 50 to Audiogenic at P O Box 88, Reading secures your entrance to the world of VIC assembly language

In touch with Aliens

Aliens have just invaded New York's West 23rd Street. They are small, dark and speak with sinister Dalek-style voices

They are not green. The Aliens are also flogging *Voice Boxes* for the Atari 400 and 800 computers at $169 (about £100) Plugged into the serial port, the synthesiser routes speech through the TV loudspeaker

You will need at least 16K of RAM to load the software, which is available on disk or cassette. The system includes a dictionary which translates typed text into *Voice Boxes'* phonetic language The dictionary can be expanded to include up to 5000 words of your own vocabulary – which may or may not be a Good Thing

The speech routines can also be called from other programs, so in no time your space invaders can be hissing sinister threats. I distinctly heard one of them mutter 'We will bury you', a threat I had previously associated only with Mr Kruschev

The Alien Group, who as previously advised, reside at 27 West 23rd Street, New York 10010 also accept Access or Barclaycard orders on 0101 212 924-5546

Heart transplant horror

Small boy to Emperor. If the 68000 is the best 16-bit CPU, why are so many people buying 8088 based computers?

Emperor: We shall require notice of that question

While the Imperial Presence consults its Civil Service, let us consider the alternatives

I mean, supposing you already own an Apple, PET or other 8-bit machine, you might not *want* to dispose of it and start again from scratch Especially if you had had a look at second hand prices lately

Happily, there is an alternative in the shape of Digital Acoustics 68000 board Fit the board inside Apples and PETs and presto! You have a brand new 16-bit computer with 92K of RAM and no loss of ability to run all your old programs as before

Digital Acoustics are an American company residing at 1415 E McFadden, Suite F, Santa Ana, California 92705 UK distribution arrangements have still to be fixed up, in the meantime they will supply direct

Like the Emperor's clothes, you won't be able to see much, but by heavens, your old machine will surely run like the devil

How to choose a computer

What is it exactly that leads people to choose one particular micro computer from the two hundred or so models currently available?

It is no idle question, particularly if you suspect, as I do, that in most cases, chance has a lot to do with it

An instant and entirely unscientific straw poll of the first half dozen micro owners I spoke to this week revealed a wealth of equally unscientific reasoning

"I liked the name" (Apple II

Plus)

"A friend had one and I thought we might swap software" (PET 4016)

"It was cheap" (ZX81)

"I had read a lot about it (ACT Sirius 1)

"I wanted to follow the TV programme" (BBC Model A)

"We already had one in our other office" (North Star Horizon)

What these responses have in common is that none of them are valid reasons for choosing a

particular micro; at least not in the sense a computer consultant would understand

I then put the question to our own tame consultant David Eldridge. How *should* you set about choosing a computer?

According to David, the key is software. 'Ask around, see who has the right software for your application. Then when you have a shortlist, try the packages out, talk to real users. It's the only way.

To that entirely sound advice I would add one caveat when considering hardware, always overspecify to take account of future expansion. It is clear from your letters that many systems that seemed ideal a year ago have since run out of steam, as file sizes have increased, and more sophisticated software has been added Especially don't be mean with memory, disk or RAM variety

by Julian Allason

Through the looking glass

*The time has come the Walrus said
To talk of many things*

Last month the trade press was full of talk of the imminent toppling of Sinclair from his throne

Of Ships and Shoes, and Sealing Wax of Cabbages and Kings

The cabbage, so to speak, who is to topple King Clive from his throne is one Gulu Lalvani

Mr Lalvani's firm Binatone are to make a wonderful new computer with colour graphics sound 16K bytes of RAM and a typewriter keyboard. All for under £50 Furthermore, it's purchasers will be blessed with the ability to run Tandy Software It would they said, smash Sinclair

Could we see it we asked Binatone

No They hadn't actually seen it themselves

Could they tell us more about it then?

No Mr Lalvani was the only person in the company who knew anything about it

Could we speak to Mr Lalvani? No

And why the sea is boiling hot And whether pigs have wings

Unipart printer

Of an evening the wizened survivors – Tommy is one of the early days of computing like to put their feet up in front of the fire Tapping out their pipes on the hearthside they tell tales of those far off days when you could plug a printer into a computer and it worked

Them days be gone says one The others nod sagely and drink deeply from their tankards

And alas the old boys are right As printers have got cleverer, the software needed to drive them has necessarily become more complex Compare say a Qume Sprint 5 Diablo 1640, Philips P2000 and one of C Itoh's TEC models and you would find they needed completely different word processing commands

The appropriately named Appropriate Technology Company of 2 4 Canfield Place London N W 6 (tel 01 625 5575) have thought long and hard about the problem The fruit of their cogitation is the RP1640 Flowriter an extremely robust, very fast (60 cps) daisywheel printer that's just that much more intelligent

The Appropriate idea is that you just plug it into *any* system and go which is a lot cleverer than it sounds

By offering the extra intelligence it has its own Intel 8085 microprocessor and 8.5K bytes of RAM memory on board – the Flowriter takes the burden off your computer, the result is much faster throughput

On top of that you have got an embryonic intelligent typewriter thrown in

This approach extends to interfaces too Just tell them whether you want RS232C IEEE 488 or Parallel Centronics and that's what you will get

I suppose you could call it the Unipart approach to computing

Cybernetic cuddle

Here's proof, or as the Legless One would say, pruf that robots are getting so realistic that they can ... well, in the interests of decorum I leave it to your imagination exactly what they can do
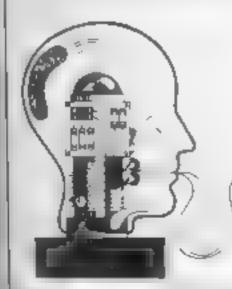
The Robot pictured here being kissed by lovely Vivien Carey rejoices in the name of Laid-Back Larry He doesn't look as if he is enjoying it, does he? Perhaps but surely not Perish the thought!

Laid-Back Larry is the property of Oxford Systems, the PETSPEED people For the uninitiated PETSPEED is the BASIC compiler that speeds up PET programs

Oxford Systems have a secret new project Perhaps it can put the twinkle back in Laid-Back Larry's eye

Comshare slashes Calc cost

Comshare, long big bananas in the world of time sharing could be expected to know quite a lot about financial modelling

So their new PLANNERCALC program for CP/M machines should be interesting The price

certainly is £39 + VAT

Like VisiCalc and the rest it is an electronic spreadsheet with a window that can be scrolled in all directions. The rules are entered in English rather than mathematical notation

It is said to run on just about every micro with an 80 column screen, 64K of RAM disks and the ability to support CP/M

The horrible object below is Comshare's logo, believe it or not

Hole in One

No This man is not Honest Joe Wozniak, Bookmaker to the Stars

He is in fact top amateur golfer, Keith Roberts, whose company ERA Consultants of Stockport, have developed a system based on the Osborne 1 portable computer, that automatically calculates individual handicaps, produces printed competition score sheets instantly, and handles golf club administration. All for £2,500 including the computer

I shouldn't be surprised if it does the washing-up as well

Canon's DIY Database cum Calc

A new microcomputer needs to be pretty special to command media attention these days, at least in the reputable journals, although there are one or two that merrily trade editorial mentions for advertising

Canon's CX-1 microcomputer isn't that innovative, which accounts for the relative paucity of press mentions. What is special about Canon is that they, almost alone of the Japanese manufacturers, seem to have recognised the importance of software (*What about Sharp? - Ed*) (Alright, include Sharp, but you're spoiling my argument - JA)

Canon have commissioned a program called CXCAT, short presumably for CX-1 CATEGORIZER. It is a programless software package that adapts to a variety of applications anything from stock control to statistics, in fact

Although it is based on a simple idea. Do-it-Yourself mini database cum Calc — it is potentially quite powerful. The essence of it is that you design

your files to match your specific data requirements. So if it is a personnel system you need, you define the file as an employee record, specifying record length, column width and titles, and the type of data to be entered in each column

CXCAT guides you through the next stage, that of data entry, with a series of conversational step-by-step instructions, allowing you to enter the data, row by row, column by column, or at random

Thereafter CXCAT organises and processes the information according to your instructions. And just as in any Calc program there is a calculation function that lets you enter a pattern of instructions, e g SALES (FIXED COSTS + VARIABLE COSTS) = NET PROFIT

Nothing revolutionary, perhaps but a potentially useful system that should garner extra sales for Canon

Gentle reader

As you may have noticed, we keep growing larger

No not me, you oaf the magazine

And unlike some of the competition, more pages means more editorial

Managing the sort of growth we have experienced over the past two years has been exciting, and on occasion, a little wearing. One of the (few) drawbacks has been the growing amount of time I have had to devote to administration not my strong point — when I would much rather have been out there exploring this bizarre micro business on your behalf

The Editor's demands for a faster secretary and a younger Porsche did it (On second thoughts that should read a faster Porsche and a younger *secretary*)

We need the resources of a big publisher behind us," I decided Choosing the right partner was the hard part

Frankly I wasn't overimpressed with most of the publishers who approached us until, that is, we were contacted by Benn Brothers a public company with equal amounts of enthusiasm, expertise and er. cash The dirty deal was

done and as of this issue *MicroComputer Printout* is published by Benn Brothers p l c

The bad news is that you lot are still stuck with our lot That's to say me the Editor, plus faster secretary, Tommy, Inside Trader, Chris Preston, Terry Hope, and all the other hooligans Fortunately for us we are still stuck with Pam and Wendy to keep the whole thing from coming totally unglued

As for the mag, well, it can only get better, can't it?

There's gold in them thar disks

Lunch Last week City of London Fat cats Cash to invest Software promising Point us in right direction Cut you in Slice of action All get rich End of story

That telegraphic tale is true The lunch took place last week It lasted four hours And me on a diet Ah well

What it proved to me is that if you have a good idea the investment is there for the asking If you have a bad idea, and it relates to software, you will probably get the money Take a look at some of the ventures surfacing in the Stock Exchange's Unlisted Securities Market (Motto Caveat Emptor)

Still the odd thing is that hardly anyone manages to consistently make money out of micro software in this country Real money, I mean

Part of the reason is that the software publishers seem to share the traditional distaste of hardback publishers for anything that smacks of marketing Check out the ads, often little better than a list of titles, with at best a few lines of impenetrable jargon

All this could change, what with Microsoft opening an office here under David Exocet' Low and three of personal computing's most creative thinkers are quite independently planning major software ventures They are Michael Healy (M D of Osborne Computers), Roger Foster (M D of A C T), and Peter Laurie (ex Editor of engineering rag, *Practical Computing*)

But while the great cogitate a small Portsmouth-based outfit called Telesoft is out there running Fast

Telesoft's speciality is the import of high quality American programs, for Atari "at reasonable prices, not the obscene levels that have been the previous going rate "

I will drink to that

Guy Kewney explains why this is the most important decision when buying a micro

HOW TO CHOOSE YOUR DEALER

It is more important that you choose the right computer dealer, than that you choose the right computer.

With a good computer and nobody to help you make it go, you are entirely on your own, and will waste time, money and sweat getting to the point where a good dealer can get you with an hour's training — at a fee of course. And a really good dealer may even be able to claim that he stocks only good computers.

The question is: what is a good dealer?

In the good old days, a micro dealer was somebody who wanted a PET but couldn't afford it. In order to afford it, the plan went, you should buy five at dealer discount rates, sell four at retail prices, and keep the fifth (free).

To get the five at dealer rates, you got in touch with Commodore (or whoever you thought made the best micro) and announced that you were setting up a dealership.

That was it.

If you were a customer, you were a freak — in those days, there were no customers. The other four machines, quite often, went to four other equally enterprising enthusiasts, who conspired with the first to get their machines at a discount.

The trouble is that today, if you are a customer, you can still feel as if you were a freak. All you have to do is pick the wrong dealer.

You can end up with an over-priced piece of outdated junk, capable of running only a few over-priced programs, which don't work until you have spent six months practising. Even then, they take twice as long to do any job as it would take you to explain it to an intelligent school leaver, because they aren't very useful.

And if you complain that the equipment doesn't suit you, the dealer will look at you with honest bewilderment. "We've never had any trouble of that sort with it," he says, intrigued by your ability to fail where so many had succeeded. He has had enthusiastic, informed, even expert customers, most of whom know more about the business and the machinery than he does. All he is there for is to collect cash for the distributors. What else?

Sheer determination

Dealers in this country, like those in America, succeed mainly because of the sheer determination of their customers to buy equipment. And the question I most dread (after "which computer do you think I should buy?") is "Who do you think is a good person to buy a computer from?"

The answer to both questions is very simple: "If you can't tell yourself, you aren't ready to buy."

Giving that answer, however, takes some nerve. I usually chicken out with some vague recommendation to go on an evening course in computing, preferably with a polytechnic where there is also a thriving computer club or Computertown UK branch.

Perhaps a more honest answer would be "There are machines which go wrong, and there are dealers who can't find out what it is — and they are in the majority of both dealers and of machines." But I can't bring myself to say that in public, because I am an enthusiast myself, and have a deplorable instinct for "spreading the Gospel" of micros.

So finally, pinned down hard, I have to admit that there is a simple checklist, and that it may help in selecting a dealer.

First and most important, the dealer must be nearby. The squeaking wheel, says the old proverb, gets the oil, and the nearer you live to the dealer, the more often you can squeak.

This is important. A truly fantastic dealer who is more than fifty miles away will turn the awesome power of his servicing on those customers who live near him and complain in person every day, and bring faulty items back immediately. You, being out of sight, will be out of mind.

Think of it from the dealer's point of view. He sells (it is Tuesday) two dozen new modem cables, to connect various computers to the phone system. Two of the cables are faulty. One goes to the man working on the next floor of his office block, and one goes to you, in the next city.

Both of you take your cables, and return to your offices.

Within ten minutes, the local man is sitting at his computer pressing buttons, and cursing. Within another ten, he's back downstairs, facing the dealer. "This modem cable doesn't work."

"It does, I've sold two dozen of the damn things. You haven't plugged it in upside down, have you?"

Abashed, he goes back upstairs, and finds that you can't plug it in upside down. Back down, he is facing the dealer again, this time with his temper in rags. "What kind of twit do you take me for? You can't plug this cable in upside down. You get upstairs right now, and get it working, or you can forget that order for a daisywheel printer."

Fast one

An order for a daisywheel printer is an order for well over £1,000, sometimes well over £2,000 if you can pull one fast enough.

Ten minutes later, the dealer is showing his customer a loose wire inside the cable. He thinks of making some comment about people who don't treat precious cables properly, but his customer's face is still a bit red. He saves it.

Later that afternoon, you get to your computer. It fails to function, so you phone the store.

"Sorry, my boss is busy with a customer, but if it's important, I'm sure he'll ring you back."

By the time you eventually get in touch with the man, two days have passed. "Modem cable?" he asks, puzzled. "We don't have any in stock, I'm afraid." It can take a lot of expensive phone call time to get things straight, and he will end up suggesting that you bring the thing back, and he'll have a look at it.

By the time you eventually get back to his town, modem cable in hand, he has completely forgotten the entire saga, hasn't a clue, and starts from square one. "Are you sure you weren't plugging it in upside down?" You have to admit that it's four days and a weekend since you tried, and you really can't be sure. Feeling an absolute fool, you drive back home.

Yes, one day you will get the thing to him and he will find out what's wrong — by then the cable will be a month old, and when he finds the broken lead, his immediate assumption will be that you've been using it for a skipping rope.

That's just with something nice and simple like a cable.

Oh, come on, let's be honest: even a cable isn't simple. You try connecting a computer to a printer yourself, and see how long it takes to get right.

Now that the importance of engineering support has started to penetrate, it is time for the second important point about your dealer. That is, he must not, on any account, be able to supply adequate engineering support.

Let me quote Colin Stanley, director of HB Computers, and now chairman of the Computer Retailer's Association, for an explanation of that apparently daft notion.

"Take a small High Street computer shop," Stanley suggests. "It is making money by selling hardware and software, and with any luck it is turning over £30,000 per month, and making a 35% margin. Work that out yourself — but it means that the guy who started the shop probably banks £100,000 in a year and probably puts half of that into his pocket. A nice wage for anyone."

So it is. The other half goes on costs — electricity, delivery, phones, and so on and on. After the first year, however, these costs start to get a bit bigger.

Installed base

"The installed base starts out at nothing, giving no problems. As it gets bigger, a proportion of all the users start needing help, needing repairs, needing advice, needing training. Our hero finds, two years after he started, that he's so busy supporting all these people that he has no time to sell machines — so he does the obvious thing. He hires a qualified engineer, and he hires a training expert."

He can't afford to be without a training person. He can't afford it because you can't afford it, when you call in with a need for training in "How to Use Supercalc" you don't want to find him booked up for the next two weeks.

The trouble is, thinks Stanley, that he fails to see that while he can't do without training and engineering staff, he can't afford to hire them, either. That £50,000 was a very nice wage but it's not much cop shared between three and there is still only one guy selling. He can't suddenly double the turnover.

Stanley reckons the answer is to set up branches, and then have a central engineering and training department. Actually, he only wants the training department until he is very big, because local engineering support still isn't possible.

If you don't believe me, watch a TV repair man at work.

He comes in response to a centrally organised service call. He takes one look at the set, and says something like (he may say this to himself): "Oh look, the old model RT#456 without sound dewobulator. Bet the grunge case is filtracicating again."

He's had twelve of these already this week. He knows that there aren't any spare grunge-case sproatbaxels, so he can't replace the faulty part. But he also knows that if he wires the flat truchorankin backwards, it will work for a month or so — by which time parts may be available, and another engineer will be able to repair it. He's done five like this in the course of the week, and none have called him back yet. Murphy alone knows why it works, but the engineering people in head office said to try it, and so far nobody's been killed.

He rewires it backwards, and is gone ten ▶

32

minutes after

By contrast, watch a computer engineer. Whatever fault the machine [......], a [...] don't have faults except yours one has [...] a certain bet he's never seen one do it before.

He's not just a repair technician like the TV man, no he's a genuine BSc Engineering with a scope, signal generator, signature analyser, and whatever he does, he can't cure the fault.

Cure it? He can't even find it. The screen which was flickering is now as steady as a rock. The disk which was buzzing like a tormented bluebottle in a can of Shandy is happily reading and writing data. The keyboard is not typing double Ts any more. And the program which crashed, [......] and printed out two pages of gibberish is now happily displaying pictures of the [...].

Proper warranty

He can't come out to your house because he hasn't the right sort of portable equipment. You take it to him whatever you think is faulty and after two days he returns it "fixed" – and as soon as it's connected up, the old fault recurs.

What you actually want is a shop which says, "We can't fix it but we can rent you one [...] work while we take yours back to head office for repair under warranty.

Third point then, you want a dealer who [......] warranty you can bet there's a reason. That

reason is, once 90 days are past that machine is likely to pack up mate.

In any case you are legally entitled to return any faulty equipment under the Sales of Goods Act. A mere 90 day warranty is not only a good example of brass neck but of questionable legal [...]. In [...] sist on a 12 month warranty [...] writing on [...] buy. And don't part with cash [...]

Finally, make sure you get it on rent before he sells. Hire a system for a month, at whatever system he asks [...]

[......] Either it doesn't have time to go [...] in which case it's what I have to [...] every day in your brother's office, a [...] you will bring the system back in a month.

Almost everything you may want to know about a dealer you will find out [...] working with him on day. He will [...] his [...] recommend certain software because certain software because [...]

[......] over the phone [......] they [...] fix your [...]

What a dealer can't do for you is [...] you want, education, software, [...] training and in repairs once you know you want it.

Anybody who [...] a dealer takes a [...] if [...] taking about [...] any off things about the [......] about [...] when you [...] after a certain period [...] there [...] [...] minutes after starting the last two weeks, don't

[......] you gave driving over the roundabout at sixty just leave him to find the [...]

[...] find a lot but not the ones you want [...]

And in the same way, if you don't know what [......] if you don't know enough about computers then don't buy one – any more than you would take a car onto the M1 [......]

With the [......] With a computer it may be your busine[ss]

MICRO SCOPE

The Operating System is a part of the computer you see very little of, yet advertisements for the new generation of computers suggests it could be the most important. So what exactly does an operating system do, and of what importance is it to the user? **Chris Preston** explains.

A question which is often asked by microcomputer users is "What exactly is an operating system and what does it do" One reason for this lack of understanding is that on a typical small home computer there is only one language available and there is no real distinction between the operating system and the language, and both tend to get lumped together as "The ROM"

However, the new generation of microcomputers will support several languages so that the presence and quality of the operating system is going to matter very much in the future which brings us back to the question, "What is an operating system?"

An operating system is really only a program, or usually a suite of programs, designed basically to take a lot of the donkey work out of writing a program for the computer it runs on. If every time you wanted to write a program you had to write your own cassette file handler, you would soon appreciate the need for an operating system!

One of the most important functions of an operating system is to present a common interface to all the individual language systems which run on the computer. Each of the language systems may well have been written by different companies, or at least by different teams in the same company. Human nature being what it is each development team is going to be convinced that their way of doing things is better than their colleagues' down the corridor and each language system is going to end up with different file structures, command syntax etc. As far as the computer user is concerned, he might as well have three different machines as far as compatibility is concerned.

File Structure

With the benefit of an operating system all the different programs share the same file structure and command syntax, regardless of what language they use. This makes life much more civilised for the user, who can buy a package written in, say PASCAL or PL/1 and then add on programs in BASIC to do some special processing he needs. We will come to more advantages later on in the article when we discuss the separate modules which comprise the operating system.

Having sold you all on the importance of an operating system, what goes into it? Well, that really does depend upon how powerful the operating system and computer are and also largely on the design. If the computer can have several people on separate terminals each running their own programs, or maybe all running the same program if they are all keying in data to the company sales ledger then the operating system needs to have modules to do a lot of work not needed in a single-user system. More of this later but we will start with a simple microcomputer system consisting of a screen/keyboard, disk drive and printer. We can draw a simple block diagram showing the various modules which go to make up the system

Command Processor	
Tape File Handler	Disk File Handler
Device Handlers (One for each device)	
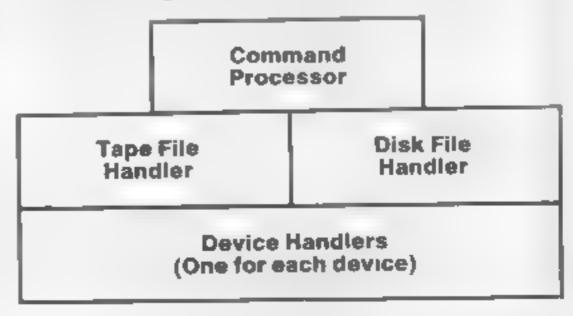
Figure 1

At the lowest level, the operating system must have a device driver for each peripheral on the system, such as the disk and printer. The device driver is the program module which actually talks to the device. In the case of a printer, its job is quite simple. The device driver will receive its data in a buffer from a higher level module in the operating system and have to send this, character by character, to the printer. After it has sent each character it will have to wait until the printer signals that it is ready to accept another character. The other device drivers will have similar functions to perform, although the disk driver for example will be more complex than the keyboard driver.

Having a single driver for a device regardless which language is in use shows another advantage of a good operating system. If each device driver communicates with higher level modules by means of buffer, then a program running at the highest level need not know whether it is "printing" to a printer, or to, say, a disk file. All the devices on the computer will behave in the same way. This facility is present to some extent on CP/M, where it is possible for example to tell the operating system to direct all screen output to the printer.

Also a device may be changed, perhaps a printer upgraded to a faster model, or a floppy disk changed to a hard disk. In our operating system only the device driver has to be changed, all the other software running on the system can remain unaltered.

The next level module in the operating system is the file handler. Again there may only be one file handler, but this is not always possible, as the file structure on a tape for example, which is a sequential device and cannot easily perform random access, will be different to that on a disk unit.

Device driver

The file handler converts the data from the format required by the user program into a form usable by the device driver. For example, a floppy disk unit reads and writes data in blocks of 256 characters, called sectors. A program running in BASIC will be writing in blocks of data meaningful to the programmer called records, which can be any length. The file handler has to "block" and "unblock" the data on its way to and from the disk unit.

According to the Programmer	Record 1	Record 2			
On the disk:	Sector 1		Sector 2		Sector 3

Figure 2

Requests from the user program will be in the form "Read record number 258" or "Delete such-and-such a file". These have to be converted into the form "Read track 15 sector 21" for the device handler. A real operating system should also be able to locate a record on the basis of an alphanumeric key such as "DTV143" instead of limiting programmers to numeric keys (record numbers). This facility is woefully lacking on most so-called business microcomputers, which means that each programmer has to write his own system, with varying degrees of success.

Notice that the command processor does not necessarily have to go through a file handler. Some devices such as keyboards, screen and backup disks may not be file structured, so the CP communicates directly with the device handler.

More powerful operating systems running on the larger microcomputers of the future will also support more complicated file structures than we have at present. If we have a file of customer information for example, certain file structures make it easier to ask the operating system, "Give

me all the customoers who are more than two months late paying".

Included in the disk file handler is a module which maintains the disk directory. The information in the directory has to tell the operating system which sectors on the disk are free, and also where all the sectors assigned to a particular file are to be found. In addition the directory may hold information on each file such as the creation date, date of last update,, whether it can be written to or not, and so on.

Above the file handler, and at the top level of our simple operating system is the command processor. This is the equivalent of the BASIC direct mode on many micros, in that it accepts commands relating to loading and running programs, displaying disk directors and copying files, for instance. This is the "user interface" and it controls the image that the user things of a "the machine". This image can bear little resemblance to the actual computer, and is often called a "psuedo-machine". The job of the operating system is to make this as simple and convenient as possible, and it is a tribute to a good operating system that the user does not realise that what he sees is not the real machine.

There is another major aspect of the operating system on large computers which with the introduction of 16-bit micros is now beginning to affect micro users. This is the

sharing of the resources of the computer between several programs or users. "Resources" in this context can be peripheral, such as disk drives, or memory and processing time.

For example, if three users are sending data to a printer at the same time, what appeared on the paper would be a few lines of output from one program, then a few from another and so on. Our poor users would need a pair of scissors and a pot of glue to make sense of their printout! Instead the operating system diverts the data from the user programs to a disk file, called a spool file. When the printer is free, one of the spool files can be sent, under the control of the operating system, to the printer, all in one go.

Obviously, not all these "advanced" features of operating systems will appear on microcomputers in the next year or so, but they have already begun to appear, and will continue to do so as the power of the micro increases first on business computers, but moving down to the home computer as time goes on. Eventually (no doubt from Clive Sinclair, costing peanuts and using half a dozen of his new flat TV screens!) we will have a home computer which Junior can use to play Space Invaders on, Sister can talk to her new boyfriend and Mum to her hairdresser over the telephone, while Father tries desperately to balance the family budget, all at the same time.

December 1979
PET in education - Survey of Business Software - Double Density Plotting - Jim Butterfield Interview - Photography Course review - The Changing Face of Commodore - Read Write - Your questions answered* - Hotline News & Products* - Pets & Pieces column* - Peeks & Pokes gossip*
Starred items indicate regular features also appearing in subsequent issues

January 1980
PET in Public Relations - A Visit to the Commodore - CompuThink Disk Drive evaluation - Survey of Programming Aids - PET's Video Logic - WordPro 1 review - Modular Programming (article & listing) - Basic ROM addresses

Feb/March 1980
Speech Synthesis on PET - HitchHiker's Guide to PET - Review - Commodore 3050 Disk Drive evaluated - PET Games report - New Approach to Subroutines - Tokens in Basic - Pet taid review - Analogue to Digital devices - The PET keyboard

April 1980
Commodore Printer evaluated - Commodore's NEW ___ ___ Assembly Language program ___ ___ ___ ___ A ___ Language Program ___ Book review ___ ___ ___ ware ___

May 1980
Personal Electronic Transactions* - Formatting numbers - Hardware Repeat key review - High Resolution Graphics Review and User Report - CompuThink 800K disk drive test - The Game of LIFE - PET User Groups - Educational Software reviewed

June 1980
PET Show Guide - 8050 Super PET - Full evaluation - Book review* - Fantasy Simulations reviewed - interview with Commodore's founder - PET ___ ___ Hardware Fair Report - PRINT ___ ___ Fund ___ ___

September 1980
Colour for your PET - Jim Butterfield's Amazing PET seminar - How to Write Better Programs - Is PET Logical? - PET has a Light Pen - Disk Lockouts and Protecting Passwords - B___ Numbering - AND, OR, NOT ___ ___ ___

January 1981
How to convert programs from ___ ___ ___ M ___ ___ ___ OZZ the best information han ___ ___ ___ ___ How to ___ ___ ___ dore's 1 megabyte Disk Drive ___ ___ ___ teach Birth Control - ___ ___ Access ___ ___ ___ ___ ___ How I developed the Stringy ___ ___
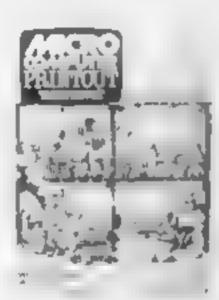
May 1981
Why VIC is the Best Home Computer yet - How to Protect Programs against Piracy ___ scratchpad program listing - Interfaces Explained - Critical Guide to Computer Books - Animating PET (with program listing) - My First Four Days with VIC - Pilot Language Explained - How to use subroutines - plus software reviews

June 1981
The VIC Report What it will and won't do - How BASIC works - PET's BASIC ___ Computer ___ ___ Complete Guide ___ ___ for PET ___ ___ Cassette Recorder reviews investigation - Approved Programs ___ ___ Plug-in chips ___ ___ How good is Comp ___ - Management System ___ New Products at the PET Show - How ___ ___ ___ - Screen Prompts ___ Significant Figures

July 1981
Could a Micro Have Caught the Ripper? - Video Games v Computer Games - Beginners Guide to Word processing - What the Salesman won't Tell You - Assembler for Beginners - Cipher Generator program listing - The Truth About BASIC ___ PET - 10 Amazing Facts About Micros - Computers in Schools - Where the Government got it wrong - How BASIC Works II - The Great ___ ___ Race game

August 1981
Looking for Lucan ___ ___ ___ ___ ___ ___ ___ for different micros - How to ___ ___ Cheap Computer ___ ___ ___ ___ cial Modelling on Micros ___ ___ ___ ___ ___ ___ ___ ___ ___ How to Buy a ___ ___ ___ ___ ___ PET - ___ ___ problems solved - How BASIC Works ___
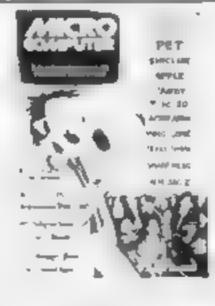
November 1981
What computers will look like in the future - PET's Screen Editor - All about Hard Disk - Programs that write Newspapers - The Truth about Computer Dating - Software Buyers Crib Sheet - Which Computer Should Buy? - Developing a games program - Peripherals for Apple - Son of VisiCalc - What use is Sinclair's 1K of RAM?

December 1981
The ___ ___ M ___ M ___ computer C ___ ___ ___ ___ Do Businesses need 6 Bits? - Converting to ___ ___ ___ most asked Beginners Questions - Bluff You Way in Computing - ___ ___ for Apple and Sinclair - VIC worth the wait - ___ Protective Game ___ Data ___ ___ the micro ___ ___ k owner

January 1982
A___ about Penny Graphics - Wide area Networking - How to buy a Business System - ___ ___ ___ ___ with a micro - D.I.Y. Business programming - Converting BASIC programs - PET Music - Apple Word Processing - Life game for Sinclair - Chessboard ___ ___ ___ Features on Osborne and S___
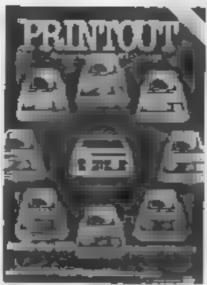
May 1982
The Computer in your Pocket - Computers in Banks - ___ ___ ___ ___ the TI-99/4A - BASIC compilers for PETs - How to buy a Purchase Ledger Program - Documentation - Energy Crisis listing - ZX81 ___ ___ - Computer fraud investigated - Anagram ___ ___ ___ ___ ___ More ___ ___ Computer jokes

Feb/March 1981

April 1981

September 1981

October 1981

June 1982

July 1982

August 1982

Binder

TOMORROW

The 5th Personal Computer Word Show this September will be the largest and most exciting microcomputing exhibition ever held in the UK.

There will be over 25,000 square feet of display space in the newest and most prestigious exhibition venue in the country.

It's the only micro computing show to offer literally hundreds of stands covering micros for home, business and educational uses. Previous visitors to the PCW Show will get some idea of the size of this year's event if we tell you that it will be approximately *three times* the size of last year's bash! (Last year's show was, in its turn, twice the size of the year before. Statistically minded persons will calculate that at this rate the PCW Show will cover the face of the planet by the year 1995.)
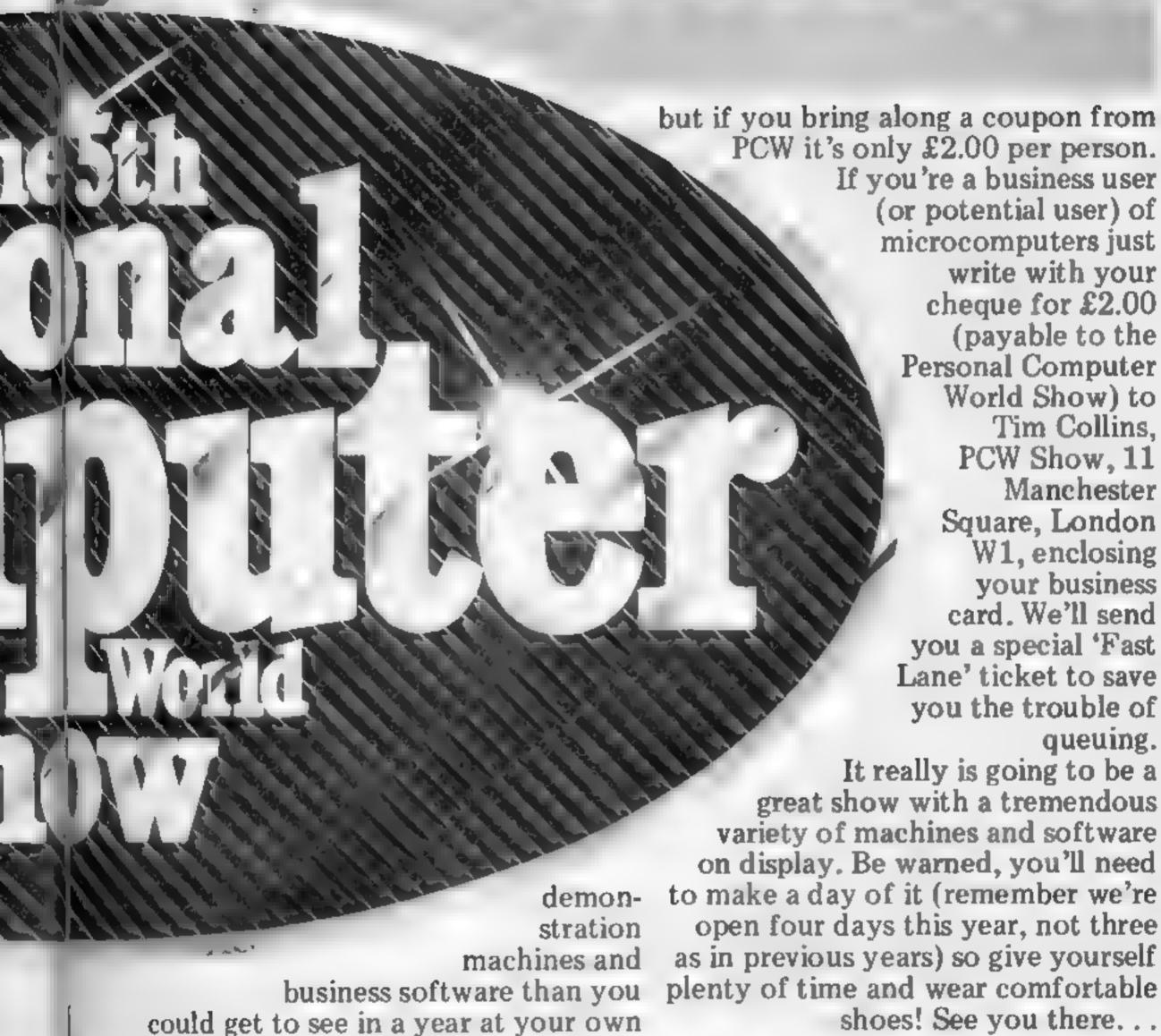
Features of The 5th Personal Computer World Show will include a Sinclair City and Acorn/BBC Arcade offering the very latest software and add-ons for these popular machines, computer chess competitions as always, and an opportunity for you to challenge a micro to a game of Computer SCRABBLE ®.

From the business angle there's free consultancy with the National Computer Centre and more

THE WORLD!

the 5th Personal Computer World Show

demonstration machines and business software than you could get to see in a year at your own office. But you needn't risk divorce to evaluate them... your wife (or husband!) and the kids can be looking at the vast array of home and educational micros in one of the other halls. It's £2.50 to get in

but if you bring along a coupon from PCW it's only £2.00 per person. If you're a business user (or potential user) of microcomputers just write with your cheque for £2.00 (payable to the Personal Computer World Show) to Tim Collins, PCW Show, 11 Manchester Square, London W1, enclosing your business card. We'll send you a special 'Fast Lane' ticket to save you the trouble of queuing.

It really is going to be a great show with a tremendous variety of machines and software on display. Be warned, you'll need to make a day of it (remember we're open four days this year, not three as in previous years) so give yourself plenty of time and wear comfortable shoes! See you there...

BARBICAN CENTRE, CITY OF LONDON
9–12 September 1982

® SCRABBLE is the copyright of J W Spear & Sons plc, Enfield

Paul Gladwell, winner of the competition, gets to grips with the Osborne 1

We present the winning entries of our 'Win an Osborne' competition, and explain how it was judged.

When we conceived the idea of our 'Win an Osborne' competition in the July issue, we had no idea of the kind of response it would generate. For those who didn't see the original entry form, we asked readers to submit a short essay describing an original application for an Osborne 1, the low-cost portable micro invented by Adam Osborne. The winner would be presented with the computer, kindly donated to *MicroComputer Printout* by Osborne UK.

The weeks that followed saw a burgeoning postbag containing an incredible variety of applications. While a few were predictable and distinctly unoriginal, the majority ranged from the creative to the outlandish. One farmer envisioned the Osborne 1 driving his tractor automatically via an ingenious network of servomechanisms and radar beacons at each corner of the field.

A serving officer from the Falkland Islands Task Force explained how the Osborne could have helped in the logistics and supply/support problems. By far the most common ideas were in the medical records and secondary education fields, with salesmen using Osbornes to prepare on site quotations, and journalists requiring portable word processing following closely.

First pass

When the Editor's In tray finally buckled under the sheer weight of entries and the sun went down on the closing date, the judges (Richard Pawson, and Mike Healy, M D of Osborne UK) were faced with the awesome task of selecting a winner. First pass resulted in a hundred or so candidates who all deserved to win, but were eventually

whittled down to thirty, then ten and finally to three. By this time the judges had capitulated into awarding runner-up prizes for the second and third places. Before disclosing the winning entries however, just how *was* the competition judged?

Well first of all we looked for *originality*. What we wanted was a specific application – not a general list of the many ways in which an Osborne could assist a small business for example. *Simplicity* helped too, since the best increase in productivity comes from doing one task well rather than several badly. Indeed, the degree to which the computer would aid *productivity* was a major factor.

We were very careful however, to avoid applications which had obviously been contrived to win. Though we considered it essential that the winning entries would require an Osborne 1 as distinct from just any micro, applications which made use of every conceivable feature of that machine just didn't ring true. *Practicality* was the key and full marks to those who had done their homework.

Limitations

You see, though the portability of Osborne had been exploited in most ideas, many did not consider its limitations. We certainly liked the idea of a water flow engineer who wanted the Osborne in a rubber dinghy – but regrettably the machine is not waterproof. Some went for the Antarctic and Trans-Sahara expedition entries. And though many had read of the shortly-to-be-released battery pack – few who made use of it had considered the 2-hour limitation and need to get to the mains on a regular basis.

Clarity of writing became important as the passes through several hundred entries wore on into the small hours. We would hasten to add that we weren't out to judge the quality of English prose (as witnessed by the fact that none of the winners were journalists). More, the judges were

looking for a clear explanation of what the computer was doing

Finally we would like to make clear that we deliberately ignored the deserving nature of any entrant. That is to say we could not unduly favour the many excellent suggestions coming from charitable bodies or those working with the disabled nor disfavour companies who could well afford a dozen Osbornes, or indeed those working in the micro industry

Congratulations & commiserations

So there it is. Obviously many of the unsuccessful entries were stronger in one particular area – and judging by the covering letters received at least half our contestants seemed utterly convinced that they were on to the winner. So in addition to our congratulations to the three winners the judges would like to add sincere thanks and commiserations to the many entrants who thoroughly deserved to win an Osborne. Better luck next time

Mike Healy, Richard Pawson

The winning entry

And now the moment you've all been waiting for! The winning entry was from Mr J P Gladwell, of 1 Leigh Way Weaverham Northwich, Cheshire and his essay on using the Osborne for land-surveying is reproduced below with the judges' comments. Mr Gladwell is pictured receiving his computer at the offices of Osborne UK and being heartily congratulated by the two judges

Modern land surveying usually consists of setting up a traverse of theodolite stations, and then taking field readings of varying distance and height differences to specific points. Each point may be a feature (e g. fence line house corner, etc), or a spot level

The field readings are converted into x-y co-ordinates and plotted on to paper. The surveyor then sketches in the detail, and the final plan is produced by the drawing office

The task of computing co-ordinates and plotting the points is one that can readily be undertaken by a microcomputer. The writer has produced a set of programs called "PointPlot" which run on a 64K micro, and interface with a precision drum or flat-bed plotter

It often happens that a surveying job takes several days (or even weeks) of field work away from base. In the meantime the office computer and plotter may be standing relatively idle. At the end of the week, there may be thousands of points to enter. Equally there are times when the surveyor could be entering data into a computer, e g when "rained off", or in the hotel in the evenings

Now with the Osborne 1, each surveyor can have a computer in the boot of his car. The additional cost is not significant. his E D M (electronic distance measuring) gear has probably cost £10,000 or more. Each evening the surveyor can enter a days work into the Osborne, and file the results on floppy disk, which is then posted back to the office (ultimately, data can be transmitted back over the telephone lines, using the Osborne's modem facility)

Back at base, a steady stream of diskettes arrive by

post and are read into the office machine and plotted out The plotter (an expensive precision machine) is kept busy and data entry is kept down to manageable proportions

Hardware requirements are 64K of RAM twin disk drives, and numeric keypad for data entry. The office machine needs an RS232 port to drive the plotter, and an extra port for a printer. Software is available, but needs revising into CP/M

Judges comments

"We were particularly impressed by the practicality of this application and its obvious bearing on productivity. Mr Gladwell has carefully not suggested that the machine be lugged around all day with the other equipment, but rather that it is used in the hotel room each evening and when weather prevents normal work. This is clearly a very real problem for surveyors

Specialist software needs to be written, but the idea has already been proven

Mr Gladwell has indicated scope for future expansion with the transmission of data via modem. Clearly, however this set up is not essential at the start

Finally, the application is sufficiently general to be of use to a large number of people

. . . . .

The second place goes to Mrs L A Woolfson of the Royal National College for the Blind Hereford, for her project with blind students. Mrs Woolfson has been sent a £100 Harrods/House of Fraser token in addition to the complimentary one year MicroComputer Printout subscription and silver Space Invader Badge awarded to all three. The latter, for the benefit of newcomers to this mag is a solid silver lapel badge designed and commissioned by us to be worn exclusively by winners of our competitions The reason they are so highly coveted is that only 30 were ever made. Mrs Woolfson's entry appears below

Visually handicapped and blind children can be successfully integrated into the ordinary school system only if there is adequate provision of classroom material in Braille. Because of the small number of children involved resource centres are being set up to serve children in several schools in the area. Both peripatetic and other teachers involved with the blind need to prepare Braille material at a number of locations home schools and at resource centres
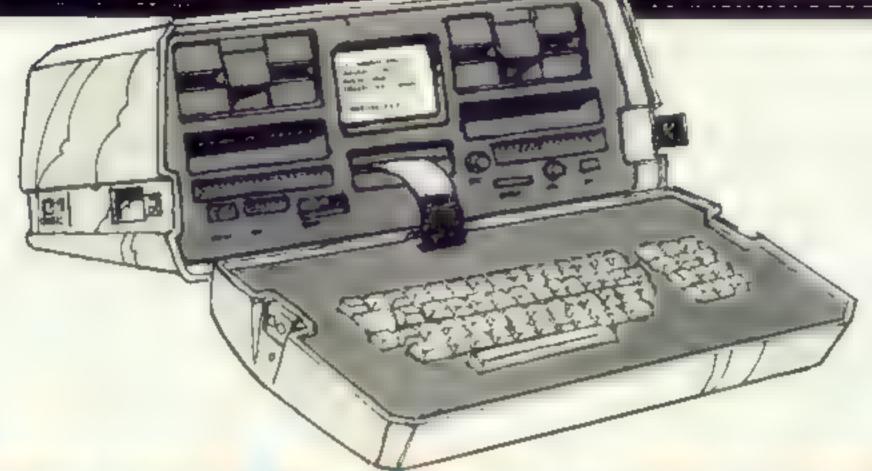
The resource centres will have Braille embossers, but two other elements are still needed. These are a suitable portable and inexpensive method for the preparation and storage of class handouts and a computer program that will convert English into contracted Grade 2 Braille

The Osborne 1 meets the first objective and the Braille translation program in its final stages of development here at the Royal National College for the Blind meets the second requirement

The Osborne 1 is inexpensive. Its operating system is CP/M for which our Braille translation program has been written. Wordstar which comes free with it, will enable teachers easily to prepare and modify text and the floppy

OSBORNE WINNER

disks will store this information. The R N C B Braille translation program will also reside on disk and when executed, will generate a further file in Grade 2 Braille. The Osborne 1 can then, because of its portability, be transported to the resource centre where, with its RS232 serial interface which comes at no extra cost, it can be connected to a Braille embosser which also has this standard interface

Braille material can then be run off, with as many inexpensive copies as required, on paper instead of the expensive plastic material normally used

Further, the original version of the text can be printed on any printer, thus meeting the needs of both sighted and blind children simultaneously

Judges comments

"Many entries concerning the disabled were received but this one deserved the most merit on the grounds of simplicity. It is not a complex one-off research project, but a practical idea that could benefit many blind young people and makes good use of the Osborne's features

Though additional (existing) hardware would be required (an example of computer-generated Braille was enclosed) the software was simple, and made use of the word processing package

* * * * *

In third place the judges chose Mr. Roger Valentine of V & H computer Services, 182c Kingston Road, Staines, Middx who has in mind a portable accountancy bureau

The unique feature of the Osborne 1 which is exploited in my 'new' application is its PORTABILITY

Actually this is not a new application at all, but a variation on a very old one. what I have in mind is a totally MOBILE accountancy bureau

With the prospect of personal computers for all possibly the concept of a computerised BUREAU may seem like a superfluous anachronism, but I believe that this opinion stems from an outmoded view of the ROLE of the bureau

In olden days, accountancy bureaux were operated by anyone with access to a mainframe, knowledge of accountancy was NOT a pre-requisite, and the bureau merely PROCESSED data, which was then INTERPRETED by the accounts department of the client. The advent of the micro has meant that many smaller bureaux have arisen along similar lines, which has brought them within the range of much smaller businesses. Unfortunately, smaller businesses do NOT, on the whole, tend to have 'accounts

departments'; either the proprietors handle their accounts themselves, or they delegate responsibility to a book-keeper. In these cases, the accountancy bureaux certainly save TIME, but with a personal computer bureau. OR manual system, it is still down to the book keeper and accountant to do with 'real' interpretive work

There are, however, a small number of SKILLED accountant/book-keepers working on a free-lance basis for small businesses. They are VERY expensive, and they can only handle at most five clients at one time (one day per week each). Given a micro, they could increase their productivity about 4-fold (2 hours per day per client), and reduce their fees accordingly, thereby becoming accessible to the smallest of businesses

Impractical? No. This is precisely what we have been doing for the past three years. The only problem we are doing the whole thing on a highly NON-portable PET. Consequently we spend most of the day driving between clients, picking up boxes of paperwork, working on them at home, and answering queries by telephone. The advantage of the Osborne is obvious. We could work at clients' premises, travelling time would be halved and problems could be discussed on the spot

Software requirements: nothing new – most of the commercial packages are unnecessarily large for the size of clients concerned, but we have written easily adaptable PET programs

Hardware requirements: again nothing really, although it would be nice to see an Osborne portable printer one day!

Cause for celebration! Shown here: Paul Gladwell (left, our own Editor (centre) Mike Healey (right), Martin 'Legless' Banks (under the table)

Judges comments

"Accounting on a micro is hardly the most original application, but here is an idea that will genuinely save time and costs for a lot of people. Interestingly, Mr. Valentine is proposing a solution for the many small businessmen who can't afford their own micro. This is personal computing as it was intended – a productivity aid to the professional, not a vast information processor."

First of all I must offer my apologies to all my faithfull readers and friends at the Commodore Show who (I hear) crammed the bars looking for me. Unfortunately, the proverbial "circumstances beyond my control" meant that I did not make it as far as London. You can ignore any rumours that my absence was due to an excess of the Editor's hospitality on the previous night though.

I did make it to the Apple Show which (coincidentally?) was being held at almost exactly the same time, not more than a mile from the HQ of Commodore UK! My first thought was that I had gone back in time two or three years. I am afraid that the show (organised by Windfall magazine incidentally not Apple UK) made several mistakes which I remember well from the early micro shows three or four years ago. All the stands were crammed together with very narrow corridors so that if you were looking at a particular product, you were continually being jostled by passers-by. And the heat!! Not as bad as the Cafe Royal but still extremely uncomfortable. Another important requirement is fairly open areas with tables and chairs for the wheeling and dealing which is so much a part of a successful show.

Moving on to the actual products on show, I was looking out for two particular ranges of products commercial programs and industrial hardware. The commercial software was very depressing indeed, being again three years behind what is now available on other micros. Basically, it could be summed up as badly thought-out, poorly integrated and *very* badly presented. Several stands seemed to have only just got hold of the software themselves, not to have much idea how their products worked, and not to really care too much. Of course I was incognito.

On the industrial side things were a lot better, at least the vendors knew their stuff, but a lot of the products seemed over-priced to me. One interesting feature of the show was the number of "Apple lookalike" computers consisting of an Apple board with a few of the standard add-ons, 80-column cards, serial or parallel interfaces and so on thrown in. These ranged from very good value to absolute rip-off so be careful!!

SYS and USR

Dear Tommy

I have several questions relating to the SYS and USR functions

1. What is the difference between the two functions?
2. Where is the numeric value of the USR function passed to?
3. The Price Device (Uncle Costan's Letters No 5) raised many questions in my mind (Dear Uncle is very interesting but bloody obscure!)

3.1 The device was primed by a SYS 826,A$,B$ command. This seems to be an example of parameters being used with a SYS command. Where are these parameters being passed to? Please explain the method used to extract the parameters (reference 150/151, $CF7B)

3.2 Where is the exit of the "Check for string" routine $CCA9 when the argument is not a string?

3.3 What mods are required to this routine for new ROMs?

4. Is a list available of addresses to interprete routines which can be used by SYS/USR commands?
J. Morton

Phew what a lot of questions! Well, a deep breath and here we go.

The SYS command, e.g. SYS 57234 just causes control to be transferred from BASIC to machine code at the address given, 57234 in this case. No parameters are passed to the assembler routine. The USR function has a single numeric argument which is evaluated by BASIC, and the result left in the floating point accumulator, 94-99/$5E-$63. This is an area of memory used by the interpreter's floating point arithmetic routines, such as ADD, DIV, DE, SIN etc. Of course, for this to be any use, you need to know where these are and how to drive them. The best publication is probably "The Hitch-hiker's Guide to the PET" sold by ACT, closely followed by "The PET Revealed", available from many outlets. I hope that one of these books will also help Mr. J. Denton-Smith who also would like to know more about the operating software.

The parameters are passed to the interpreter's parsing routines which are called by the Device. When the SYS call is executed by the interpreter, the test pointer used by BASIC is left pointing to the comma just after the 826. The "Check for Comma" routine checks that you have in fact put a comma in and not a question mark or something. The next routine parses the variable you have specified and returns lo pointer to the BASIC variable. The "Check for String" routine makes sure that you had given a string variable. If you had put in a numeric variable by mistake, you would get the standard BASIC "TYPE MISMATCH ERROR". So after parsing all the arguments, you are left with two pointers, one to A$ and one to B$.

So what comes next? This is the format BASIC uses to store the details of a string in the variable table (the data itself is held in the top of memory), and is called a dope vector.

Length	Address (low byte)	Address (high byte)

To swap the strings, all we need to do is swap over these two vectors. BASIC has to do a lot more work, which involves actually moving the data itself around, because PET BASIC, unlike some other languages, lacks a SWAP instruction.

Here is a short conversion table for the "Price Device"

	Old ROM	New ROM	BASIC 4
Check for comma	$CE11	$CDF8	$BEF5
Parse Variable	$CF7B	$CF6D	$C128
Check for string	$CCA9	$CC90	$BD89
First pointer	150/$96	68/$44	68/$44
Second pointer	176/$B0	94/$5E	94/$5E

I hope that this has sorted out all your problems. Although it is not really practical for us to publish pages and pages of ROM disassembly, if neither of the two references above can help, why not drop me a line?

The dastardly DS

Dear Tommy

I have access to 3008 and 4008 PETs. I have written a program which runs perfectly on the 3008 but continues to crash on the 4008 at line 190 which reads

190 SD=DS+A   DS=0

The only conclusion I can arrive at is that either SD or DS must be a reserved word in BASIC 4 but not in BASIC 2. Is this correct? If so, what is its function and are there any similar reserved words used by BASIC 4 and not BASIC 2 which I should look for?
B. Pierce

You've got it! DS is indeed a reserved word in BASIC 4.
Together with DS$ it is used to interrogate the disk status.
Under BASIC 2 to find out if a disk error has occurred, you
had to do an INPUT #15,A$,B$,C$,D$ which read the error
number, name, track and sector into the four variables. With
BASIC 4 things are a little easier. DS$ reads all the status
information into one string, DS gives just the error code, so
you can test for a disk error quite simple

100 IF DS > 0 THEN PRINT "DISK ERROR" : END

DS and DS$ are the only new 'variables' which are
reserved under BASIC 4

Point of no return
Dear Tommy
*On my PET 4032 I have developed several programs which
wait for an answer to a Y/N prompt controlled by the GET
statement, e.g.*

10 GET A$: IF A$ <> "Y" AND A$ <> "N" THEN 10

*If I follow this with an INPUT statement, how can I prevent
the program terminating by a mistaken pressing of the
RETURN key after the Y or N?*
A.J. Harrison

There are many ways of avoiding this, here are just three to
be going on with. The first two work by throwing away any
characters which the operator has keyed in, and can be
stuck in before the INPUT statements

100 GOSUB 1000 : INPUT "ENTER YOUR NAME",A$

Firstly we have

1000 GET A$: IF A$ < > "" THEN 1000
1010 RETURN

This is a straightforward routine which just reads every
character out of the keyboard queue so that you start the
INPUT statement with a clean slate
A little more subtle is this

1000 POKE 158 0 : RETURN

This has the disadvantage that it may not work on a future
release of BASIC or on a different machine.
The third method is to stop the INPUT statement
aborting the program when nothing is entered. The simplest
way to do this is by giving the INPUT statement a special
prompt, instead of saying

INPUT "ENTER YOUR NAME",A$

you will have to PRINT the prompt. For the INPUT statement
prompt you must enter 3 shifted SPACE characters and 3
cursor-left characters. Now the INPUT statement will not
stop the program if you just press RETURN, and you can
test for this by saying

100 IF A$ = CHR$(160) THEN

Once again there is a sneaky way of doing this, which was
pointed out to me by Paul Hooper of Liverpool. The INPUT
routine in the interpreter is also used by the INPUT#
statement, which in effect just changes the input device
from the keyboard to, say, the cassette unit, then calls the

INPUT routine. The INPUT routine uses a zero page
location to test if the input is from the keyboard or
another device, and in fact this is one of the few zero
page locations which is different between BASIC 2 and
BASIC 4

100 POKE 16,1 : REM POKE 14,1 FOR BASIC 2
110 INPUT "ENTER YOUR NAME ? ",A$
120 IF A$ = "" THEN 110
130 POKE P,0

There are two things to watch out for. Firstly, the INPUT
routine will not now print the question mark after the prompt
string, so if you want one you will have to include it in the
prompt string as above. Secondly, the INPUT routine will not
print a carriage return after the INPUT statement, so you
may have to put in some extra PRINT statements to keep
your screen layouts OK

Colourful language!
Dear Tommy
*Being the owner of a 40-col PET who would like colour
output on the screen, but not wanting to purchase a VIC
and rewrite my software, I wonder if you could help with the
following questions*

*1 Will any of the chips from the VIC (e.g. the character
generator) work in the PET? (Assuming a colour screen has
been used to replace the present one.)*
*2 If not, will those in the VIC 40 (which is apparently more
compatible with the PET than the VIC 20) be useable? I am
not interested in the graphics at the moment, just normal
text output*
*3 As I already use the user port, is there any way to use the
VIC's lightpen, paddle and joystick on my PET?*
R C Pickup

I am surprised that nobody has written before about this.
Unfortunately, there is not much joy for you. The VIC video
system is built around the Video Interface Chip which gave
it its name. The only way to use any of the VIC circuitry in a
PET is to go the whole hog, rip out all the PET video circuitry
and replace it with the VIC system, rewrite the ROMs
accordingly, and rewrite all your programs to use your new
22 column screen! In short, as I said — no hope
Similarly, the VIC lightpen and paddles use the VIC
chip, and cannot be used on the PET. The only chance you
may have is with some joysticks, which use four digital
switches instead of having two analogue potentiometers
mounted at right angles. These could be connected to the
user port and tested for by a PEEK statement
On the other hand, were you to purchase the VIC — or
the new Commodore 64 — I think you would find it
surprisingly easy to rewrite and run all your PET programs

Stacks of memory
Dear Tommy
*My CBM 8032 is giving me an "OUT OF MEMORY"
message after a dozen or so passes through a record
keeping program, with files on disk, that I have written
myself*
*My Toolkit shows me that the error is occurring in a
FOR NEXT loop and the manual tells me that since FRE(0)
shows plenty of memory available, this is probably because
the stack is full*
*I have little or no understanding of the stack, so what
can I do about it please?*
P Blake

▶

TOMMY'S TIPS

The stack is the area of memory BASIC uses, amongst other things, to store FOR-loop and GOSUB information. It is only 256 bytes long, so it is not too difficult to fill it up if you are not careful. The usual problem, however, is a bug in the program rather than a program being too big or complicated. The most likely cause is a GOSUB without a RETURN, which means that each time you call the subroutine, you use up extra stack space which is never freed by a corresponding RETURN. Check all your subroutines to make sure that there are no possible exits by a GOTO rather than a RETURN; especially error and except on handling exits. It is good practice only to have one RETURN at the end of the subroutine, and if you want to leave the subroutine halfway down, so to speak, GOTO the one and only RETURN. The other possibility is that you have simply got too many active FOR-loops. Are you ever just jumping out of a FOR-loop without finishing it off? This is the only safe way to leave a FOR-loop.

```
100 FOR I=1 TO 100
110 IF A(I) < 0 THEN I=100 : REM SEARCH FOR
    NEGATIVE ELEMENT
120 NEXT
```

Finding these bugs can be a great problem once a program is written, it is far better to get into good habits from the start.

Long playing record

Dear Tommy

Some time ago I got hold of a secondhand Apple II with various extras, and set to to teach myself how to use it. I recently got as far as using random disk files, and I must admit I got into a pretty mess at first. I think I have got the mechanics sorted out but I cannot understand how the data is actually held on the disk, and what happens if the string written to the disk is not the correct length? Why do you have to specify the record length in the OPEN command? Cannot the DOS get it from the string you write to the disk? Also, what is the use of the B parameter when selecting a record number?

W P Johnstone

It will help if you think of the file, not as it's recorded on the disk, but as a straight line.

```
START
BOX
BUSE
BYTE
```

RECORD # 1	RECORD # 2	RECORD # 3	RECORD # 4

Figure 1

1   21   41   61   81

Record length = 20 bytes

DOS calculates that the record starts at byte $20*(n-1)+1$

If you now tell the DOS to write to record number 3, it has to find out where the record is, and it can only do this if it knows the record length. Some operating systems hold the record length in the disk directory, but with the Apple system you have to give the length each time you open the file. Having found the record position (and the byte within

---

GOO

A look at

Under growing pressure
from unions, computer
manufacturers are now
placing as much
emphasis on the exterior
design as the workings.
James Woudhuysen,
former editor of DESIGN
magazine, explains what
'ergonomics' is all about,
and how to spot a well
designed micro.

One of the very few growth sectors of the
international publishing industry these past
few years has been that which covers the
ergonomics of computer terminals. Even
back in 1980 Colin Mackay, senior
psychologist at the Employment Medical
Advisory Service of the Government's
Health and Safety Executive could write a
12-page booklet entitled "Human factors
aspects of visual display unit operation"
(HMSO, £1.50) and include no less than 57
different references on the subject. Parallel
with the rise of scientific interest in computer
ergonomics has come the rise of commercial
attention to the issue. More and more
computer manufacturers have stressed the
'user friendliness' of their machines

Things have got so sophisticated in the
mainframe sector that manufacturers such

as Olivetti, Philips, Nixdorf, DataSaab and
Siemens now put out special ergonomics
manuals with their machines. But what
about the ergonomics of microcomputers?

Of course, the question every
microcomputer user would like answered is
simply 'what constitutes good ergonomic
design on a micro?' But if you ever get a
snap reply to this you can be sure you're
talking to a marketing man, not an
ergonomist. An ergonomist will tell you that
the question is a very complex one, that it all
depends, and so on. And, frustratingly
enough, he'll be right

Good ergonomics on any computer is
first of all about what you want to do with it
something that varies a lot from user to user
True, there are some ground rules on
keyboard and VDU design, and there are
also some suppliers who can be considered
as ergonomic leaders, not only in
mainframes, but in micros too

But before going into these aspects of the
problem a quick look at what ergonomics is
and how computer ergonomics first
grabbed the limelight

Ergonomics means a lot
Roughly translated from the ancient Greek
ergonomics means 'legislation for work'. As
a discipline, it was born in the Second World
War when new weapons technology, the
need to use (and train quickly) conscripts
long hours of anxiety and boredom and
short bursts of intense concentration, all
conspired to make research into
'occupational stress' essential. Britain has
always been good at it (the National Coal
Board's ergonomics department is one of
the largest in the world), the Soviet Union
takes it very seriously (Soviet tank interiors
are reputed to be ergonomic masterpieces.

and the number two in Soviet design is an
ergonomist of international standing), and
America has a flourishing academic and
corporate culture in what it calls human
factors
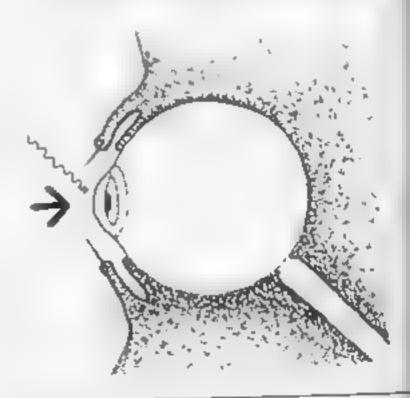
Which brings us to an important point.
According to Dr. Ivan Brown, of the Medical
Research Council's Applied Psychology
Unit, Cambridge, ergonomics (the British
term) used to be more about anatomy and
anthropometry while human factors (the
American term) used mostly to be about
applied psychology. Now, however, the two
terms are roughly synonymous, both
ergonomists and human factors experts
acknowledge that, as far as human
behaviour is concerned, mental and
physical processes are indivisible, if only
because modern technology has become
so taxing to work with

In this article we shall, purely for reasons
of space, deal with broadly physical
problems – we'll leave the more cerebral
ones to another issue. But the 'broadly' in
the past sentence is there for a purpose.
Keyboard layout, layout on screen, colours
on screen, etc. are all matters which directly
overlap both hardware and software
domains. The designers of Xerox's 8010
Star Information System office personal
computer, for example, have gone on
record as saying that they had to do 30
man-years of research into 'where the user's
head was at' before beginning hardware
development. So don't imagine that
programming methodology doesn't have
anything to do with musclepower, because
it does

The unions started it
Where, however, do all today's obsessions

OD DESIGN

at Ergonomics on micros

with computer ergonomics spring from? Well, from the reactions of users – unionised users.

Back in the mid-seventies, Scandinavian and West German trade unions became the first to wring concessions out of the employers on computer installations, and today Denmark, Sweden, Norway and West Germany are the countries in which legislation on terminal design is most advanced (it's a statutory requirement in West Germany, for instance, that keyboards be equipped with pads for operators to rest their hands on). The reasons for this state of affairs are simple enough: the four countries were amongst the first to invest in computerised offices, and they were and remain countries in which union participation in the planning of new investment is particularly well-established – in contrast to France or the USA, where looser industrial relations environments obtain. However, the reasons the unions made a fuss about ergonomics were far from simple.

In Denmark and Sweden strikes took place over the ability of sophisticated terminals to monitor worker performance (counting the number of keystrokes made an hour was one of the favourite techniques for doing this). That was enough to warn managers of all nations off crude, 'if you don't like it, jump it' approaches to the introduction of new technology, so other issues tended to become more prominent instead. These were, roughly in order of discovery: radiation hazards, visual fatigue, epilepsy and overall workplace design. In other words, 'good ergonomics' tended to become a more and more subtle business as the unions became more and more experienced in dealing with the gadgets

their members were being asked to work with.

Ergonomics is not, therefore, a wholly technical, 'objective' thing – it's social, too and because society changes, it changes. Today VDUs have been given a clean bill of health as far as X-ray, radio frequency, microwave and ultraviolet emissions are concerned, but unions are still agitating about them. While union membership in the white collar sector has expanded, the rise in monetarist attitudes among management has meant that a big gap has opened up between white collar union size and white collar union influence. In 1982, white collar unions see in computer ergonomics a relatively 'soft' means of getting consulted and thus recognised by management.

The International Federation of Commercial, Clerical and Technical Employees (FIET) published *Ergonomic problems with visual display units in the banking and insurance industries* in 1976. Within three years chapters on computer ergonomics appeared in APEX's *Office technology: the trade union response*, the NUJ's *Journalists and the new technology*, NALGO Action Group's *New technology* and the Socialist Workers Party's *Is a machine after your job?* The interesting thing about these pamphlets was that whether moderate or militant in tone, they

* said roughly the same thing about computer ergonomics, mainly because they used the same 'establishment' sources on the subject
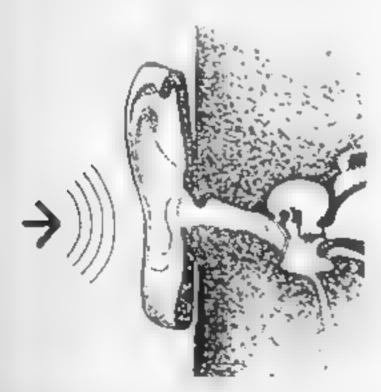* overestimated the extent to which computers would be introduced into British offices, given widespread financial stringency
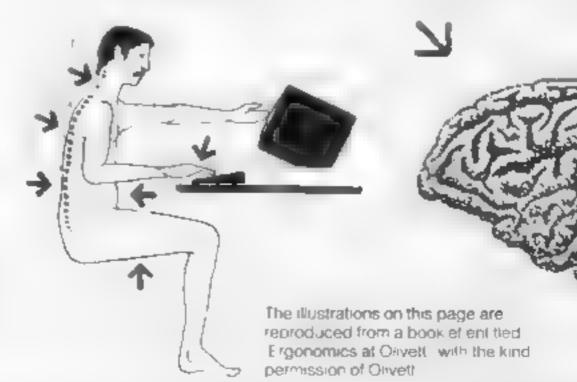* conversely, underestimated the extent

to which those few companies which could afford to introduce computers in a big way would be prepared to spend money on ones guaranteed to go down well with staff
* underestimated the speed with which computer-makers would adjust to the market requirement for good ergonomics

In fact what has happened is that, as a consequence of union pressure, the eighties have seen most of the major (physical) ergonomics problems directly associated with mainframe terminals solved. Today, computer ergonomists are more interested in the interaction of terminals with their surroundings than with keyboard or VDU construction. George Sowden, British-born senior office products designer at Olivetti, Milan, sums up: 'Just as the first motor cars looked like horseless carriages, so terminals started off looking like typewriters-and-screens. But once the technology had matured a bit and people had realised that because no one operator position is best, flexibility is all – then brains, keyboard and screen went their separate ways. After that black net filters came in as a way of minimising extraneous reflection ("glare") on screen, and manufacturers started to crank refresh rates up to fifty hertz or more to avoid bringing on epileptic fits. There are still "intrinsic" problems with terminals: very black, glareproof screens tend to make characters look like torches in the night, and new high definition technology is only just coming on stream. But most major manufacturers have got it right. Olivetti recently supplied Abbey National with computers to the tune of twelve to fifteen million pounds, and it obviously had

The illustrations on this page are reproduced from a book of ours titled *Ergonomics at Olivetti* with the kind permission of Olivetti.

Different strokes for different

f you're a teacher buying micros for your class the three separate units (computer keyboard and VDU) that some micros come as could mean three times as many accidents or ripoffs at school You'll need to ensure that the manual that goes with your micro is big enough not to get lost easily, and that it will withstand rough treatment ruggedness is vital on a school equipment

If the keyboard on your micro is going to be used by secretaries or journalists they may find its differences irritating in comparison with the boards they're accustomed to If on the other hand you're going to spend most of your time playing games or juggling numbers on your micro the precise alphanumeric layout on it isn't going to bother you too much if you're bilingual and into words you might want to

to be able to promise the client that its machine wouldn't prompt union resistance The real ergonomics matter these days is the environment

Fine The snag is, though that while unions may have forced mainframe terminal manufacturers to sharpen up most microcomputer users are not unionised. So if you want to buy a micro with 'good ergonomics', you've got quite a job on your hands More fundamentally tasks at a micro often differ from those at a mainframe terminal Before you go any further, then you should establish just exactly who this micro you want is destined for

hunt around for a model which can switch the characters activated by its keys from one language to another

f you've got kids, you might want to lash out on NECs PC 8001, which can turn a whole QWERTY board into an ABCDE one f you're after a compact desktop or pocket machine you'd do well to remember that the size of its keyboard will make repeated use a hassle unless your hands are particularly delicate and you'd do well to remember too that the cramming of lots of functions on to each key has its disadvantages (unless the manufacturer has arranged for you to be told what 'mode you're in very carefully preferably by giving your cursor different shapes you'll find yourself making mistakes) And if you're buying a micro for somebody as a present and you feel generous enough to pay for colour graphics check that somebody isn't colour

blind getting on for one man in ten has
defective colour vision

Get the picture? A lot depends on you.
Take VDUs, is the VDU you want going to
be used in a factory by any chance? If it is,
make sure it doesn't pull in cooling air over
its screen and so make viewing a dirty
business. Is it going to be used in American
banks (which are very brightly lit) or in
Scandinavia winters (in which the sun barely
rises above the horizon)? Alternatively take
pointing devices. If you must point, you'll
find that cursor step keys are too slow and
that they don't work with graphics, but if you
want to point on board your racing yacht
you might find a light pen a bit hard to
handle. Your best bet is probably a 'mouse'
a sensor based device you roll around on
your desk to tell your cursor where to go. It's
handy and soon becomes an extension of
your arm, though few companies offer such
a device.

The best way to set about buying your
micro is to think about what you're going to
use it for. And when you try one out, follow
the kind of routines you'd be using if you
bought it – don't follow demonstration
programs. Needless to say, your most
successful gambit is likely to be finding a
friend who has exactly the same
preoccupations as you and who has already
got a micro that works like a dream.

Clarifying your context

Sorting out who you are and what you want
to do only requires intelligence. Sorting out

the environmental context for your machine
requires either DIY skills and a healthy bank
balance (if you're computing at home), or a
fair amount of clout with your office service
manager (if you're computing at work). The
recommendations below apply to mainframe
terminals under heavy operator load
because working with micros isn't generally
as arduous as this, you'll be doing very well
if you manage to fulfil half of them.

Lighting Illumination levels of 400 lux are
about right. Much lower than this makes for

GOOD
DESIGN

high contrast on-screen but an atmosphere
which is depressing and soporific, while
higher illumination levels make for much
higher illumination levels make for much
lighter overall but glare.

Glare, shield overhead lights with grills,
make sure they run parallel to your line of
sight and put them forward and about a
metre either side of you. Avoid the
combination of dull background and bright
spots on source documents. Line the side of
your VDU up parallel to your windows and
put a screen behind you. Go for matt
finished grey furniture and buy yourself
some roller blinds, or, better, those trendy
looking curtains that come in the form of
separate, buff-coloured strips of fabric
strung out in lines. Set contrasts between
screen, documents and background in the
ration 1 3 10

All the machines featured here have
been noted for good design in the
press. Left hand page from top
Compass Grid, Corvus Concept,
Hewlett Packard 125. This page DEC
Rainbow, Olivetti M20

Model	Manufacturer	Designers
Commodore 64 Commodore II	Commodore	Porsche, Zell-am-See, West Germany
	Convergent Technologies	ID Two, Palo Alto, California, consultants, Mike Sanders et al in-house
Rainbow	Digital Equipment	In-house team
Compass computer	Grid Systems	ID Two, Palo Alto, California
HP 125	Hewlett Packard	In-house team
IBM Personal Computer	IBM	In-house team
PC-8001	NEC	In-house team
Nexos 2200 word processor	Nexos	Satherley Associates, London
M 20	Olivetti	Ettore Sottsass, Tony Maccia-Cass Man
ACT Sirius 1	Sirius	Peter Senhal et al in-house
Series 6000	Racal	Crisp & Wilson, London
8010 Star Information System	Xerox	David Canfield Smith, Charles Irby, Ralph Kimball, Bad Verplank, Eric Harsiem (in-house team)

Humidity, room temperature, noise
Install air conditioning if you can afford it but don't let it make the room draughty. Don't let the room get hot and dry – especially if you wear contact lenses. Soundproof or if possible remove any peripheral equipment (especially if it gives out more than 65dB). Put carpets, curtains, makeshift baffles (bookshelves, cupboards) and above all three-dimensional ceiling coverings everywhere to absorb noise this is particularly necessary in open-plan offices.

Posture, work organisation, etc
Invest in a good office chair. This is essential. There is a survey of some of the models on the market in the April 1981 issue of *Design* magazine. The angle and height of both seat and backrest should be adjustable especially if more than one person is going to have use of them (which is why quick-action, gas-operated chairs make a lot of sense). Seat height should be about 40cm or more, the backrest should be 50cm long and angled back a good 20° from the vertical. Adjustable desks (including electrically adjustable ones, such as those recently brought out by ITT) are available; try to get a thin one 70cm high or 17cm above your knees when they and your hands are in a horizontal position (which is the position they should be in). If you're small you may need a footrest.

Learn to touchtype, put a good 50cm between your eyes and the screen, make sure that the screen is set at right angles to

your line of sight, and place it 20° below the horizontal. Buy an adjustable document-holder that holds sheets 5cm above the worktop; place it next to the screen at an angle of 20° to the vertical, or, if you find you're changing documents frequently, site it between keyboard and screen, to one side, and at an angle of 65° to the vertical.

Maximise desk storage space and keep wiring out of the way. Put both VDU and documents within a 35° arc around your line of vision. If you're doing a lot of keying using 10 fingers, place your keyboard parallel to and 7cm from the edge of your desk. If you're flipping through documents and entering a lot of figures, put your documents at a slight incline to the desk, shift the keyboard to the right (presuming you're right-handed) and use your hands independently. If you're doing a little keying and a fair bit of writing at the same time keep your documents flat, use both hands independently and splay both documents and keyboard 17cm away from the edge of the desk. Make sure your clear working area is at least 60cm wide and 90cm deep.

Personal tips: take frequent rests, they can be of short duration, but take them before you get tired, not after. See an optician – preferably one who knows about VDUs, and, if possible, one who's prepared to come and inspect your set-up. Remember that reading glasses aren't suitable for the half-metre distances you'll be working at. Try to work from high contrast, high readability, high legibility documents. Decorate your room with soft, dark colours

unless you're doing a lot of repetitive work in which case soft, cold colours and a few dashes of brightness are best. Try not to drink too much, and lay off the fags, valium etc too.

The right kind of micro
Now for the moment we've all been waiting for – the ergonomics of the micro itself. Once again the recommendations here apply to mainframe terminals in the first instance and micros only in the second.

Keyboards, they should be detachable with thin, flexible cables of reasonable length (keyboards that work via infra red remote control do exist). They should have some facility for attaching strips of paper or card with function identification on them. They should be easy to move about intentionally and hard to move unintentionally (footpads are useful here), have their home keys 3cm above the desk, and be inclined to the horizontal by an angle of about 10°. Like everything else on the terminal, their surrounds and their keys should be matt-finished in grey. Key rows should either be stepped or sculptured so that the upward curvature of the board follows the radius of your fingers. For number-crunching applications they should be supplemented by a numeric keyset. There should be some acoustic feedback to let you know when you're stroking, and tactile feedback should be arranged so that the pressure and depression needed to make an input are greater than those needed to release a key

GOOD DESIGN

The VDU and the display itself you should be able to tilt and swivel the VDU at will, and you should be able to vary its height by at least 10cm. Phosphor persistence should be high to minimise flicker. Screens can be any colour, though most people prefer yellow phosphor and amber filters. Refresh rate should be constant and as high as 80hz if possible. Bit-mapped displays — in which every dot can be individually turned on and off by memory — are the best way to improve definition (the alternative interlacing scan decreases refresh rates) and are essential for the reproduction of multiple fonts and for variable line spacing.

Glare is best reduced by black net filters at illumination levels of more than 500 lux, or by a toughened CRT at levels below this. Character luminance should be adjustable but range around 150 nits, screen luminance should be a good 10 nits.
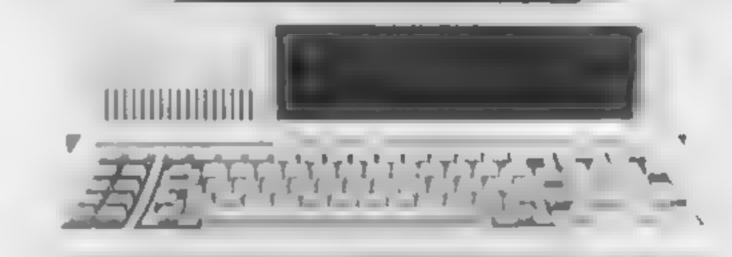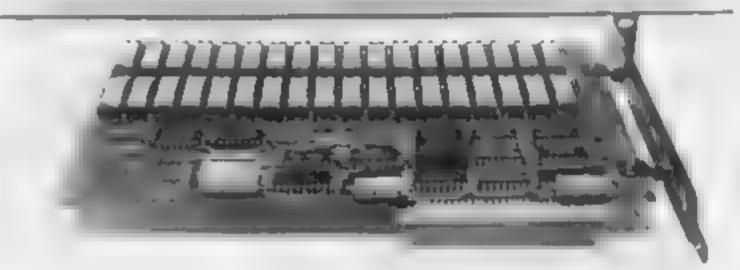
The ergonomic leaders

So who are the micro manufacturers whose machines begin to measure up to the specification listed above? Peter Senhar, Vice-President and co-founder of Sirius Systems Technology, Santa Cruz, California: "The recent arrival of third generation, 16-bit micros from companies like Sirius, Digital, Olivetti and IBM has been accompanied by a new emphasis on the needs of the continuous users, whether he be fanatical home hobbyist or normal ground-down office worker. I admire the ergonomics on all those firms' machines. But if you go to second generation, 8-bit micros such as those made by Apple, Atari, Texas, NCC, Toshiba or Hitachi, you're really talking about toys."

In other words, the most ergonomically satisfactory micros are those which are most up-market (surprise, surprise). Here's a list of some of the front running models

the firms that make them and the design teams that put them together.

Conclusion

Early on in this article I pointed out that ergonomics is a growing thing. That fact suggests one conclusion and one

market forces will see to it that a firm's investment in good ergonomics is rewarded directly by sales in the shops.

Even highly used 8-bit micros can be upgraded considerably at little extra cost. It's time micro users started voting on ergonomics — voting with their feet.

Free with every Commodr
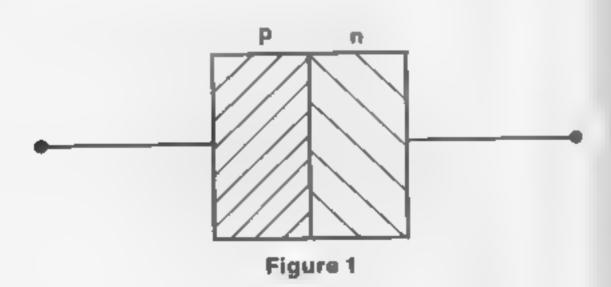
HARDWARE

If you open the lid of your computer, you will see a large number (or possibly not a large number if it is made by Uncle Clive), of black objects known variously as ICs, chips or beetles. We have already looked at some of the smaller ones, which are called gates, but in this article **Chris Preston** investigates some of the larger devices which go to make up a computer.
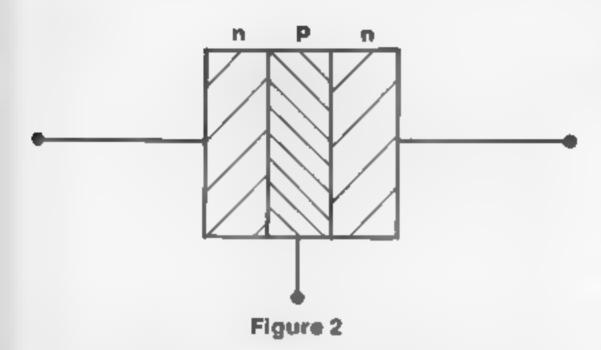
One way to think of the chips is a labour-saving device for people in general. They perform a lot of functions which can be done in software, but of course they work a lot faster than any program.

Before we look at chips and data, it would be interesting to see how these devices are actually made. We can start by looking at the simplest semiconductor device, the diode. The diode is a component which conducts electricity in one direction only. If we take a piece of extremely pure silicon and add a certain type of impurity, we get what is known as p type silicon. Adding a different kind of impurity gives us n type. Put the two together and we get a pn junction, forming a pn junction diode.

Figure 1

By sandwiching a piece of p type silicon between two pieces of n type, we get an npn transistor.

Figure 2

Transistors and diodes can be connected together to form any of the gates we have previously talked about.

In practice though, the components are not made by joining different pieces of silicon together, but instead by a process called diffusion, where we start off with a single piece of silicon and build up a pattern of impurities which produce the circuit we want. By varying the pattern we can build up a circuit – in just the same way as would be possible using discrete components soldered together with pieces of wire. We can in fact produce almost all the components used in the electronics industry, but some are easier to produce than others.

How do we control which impurities go where? Well, the process used is very similar to that used by an engraver. He covers his piece of metal with a substance called a resist which is not affected by the acid he is using. He then draws a picture by scraping away the resist to expose the metal underneath, which is then attacked when the engraver immerses the plate in the acid.

When an integrated circuit is manufactured, the surface of the silicon is covered with a "photoresist", which is sensitive to light just like a photographic emulsion. The manufacturer then exposes the resist to a light shining through a "mask", which is a negative of the pattern he wants on the silicon. The parts of the photoresist which have been exposed to the light are chemically altered and can then be removed with a solvent, which reveals the silicon underneath. The whole piece of silicon is then exposed to the impurity required, which diffuses into the exposed areas on the chip. By repeating this with different masks and different impurities the manufacturer can build up the circuit he wants.

There are, in fact, many other ways of introducing the required impurities, and other processes involved in the manufacture, such as connecting wires between the chip and the outside world, but you will have to make do with that for now if we are ever going to get round to looking at some real chips.
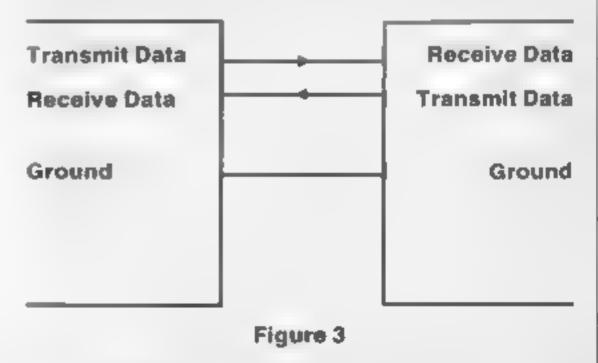
Communications Chips

Back in the bad old days, if you wanted to get two computers to talk to each other, you had to build an interface, which could be a box the size of a present day microcomputer. The introduction of large-scale integrated circuits has meant that the entire interface is now built on a 24-pin chip which can be tucked away inside the computer. This means that most microcomputers either have comms facilities as standard or as a fairly cheap add-on.

The most common standard for communications at the moment is RS232, although there is a good deal of variation even here. A new standard has just been announced, the RS423 which offers great advantages over RS232, such as increased speed, increased distance between stations, and better noise immunity. Already one microcomputer, the BBC micro, has an RS423 interface and it looks certain to be very popular in industrial applications where electrical noise from machinery is a problem.

Because RS423 is not yet terribly common, we will only look at the RS232 standards here. Two typical chips are the 6850 (Asynchronous Communications Interface Adaptor or ACIA) and 6852 (Synchronous Serial Data Adaptor or SSDA) from Motorola, although just about every major chip manufacturer produces a communications chip of some sort.
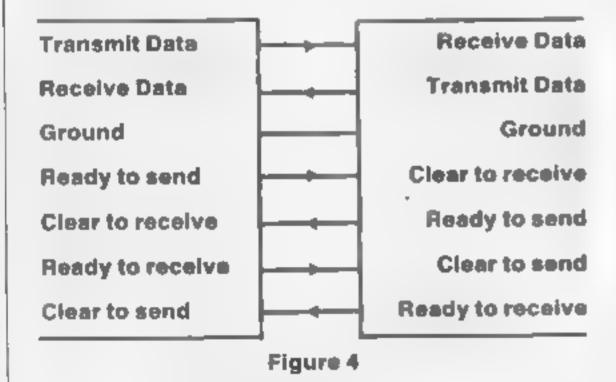
The job of the comms chip is to convert bytes of (parallel) data from the computer into a string of (serial) bits which can be sent down a wire to a peripheral or another computer, and to receive serial data from the other station and convert the data back into parallel form for the computer. The simplest form of RS232 interface is called a 3-wire interface, because it consists of just 3 wires. Two wires carry the data, one for transmitted data and the other for received data, while the third is just a common or earth line.

Transmit Data		Receive Data
Receive Data		Transmit Data
Ground		Ground

Figure 3

The problem with the simple 3-wire system is that the receiver may not be ready to receive. If it is a printer it may be

In addition to all this, the comms chip can also be programmed to do a certain amount of error checking using a system called "parity". If we take a byte of 8 bits, and count up how many "ones" there are, we will get either an odd or an even number, and so we say that the byte has either odd or even parity accordingly. If we are using a parity checking system, we will add an extra bit, making 9 in all, so that the group as a whole always has the same parity. The receiver checks the parity on each group of nine bits it received, and if it finds that the parity of the group is wrong, then one of the bits in the group must have been corrupted during transmission. There is a bit in the status register to tell the receiving computer that a parity error has occurred. Of course, if two bits are wrong then the parity will be right again but you cannot expect perfection from such a simple system, and two-bit errors are much rarer than one-bit errors.
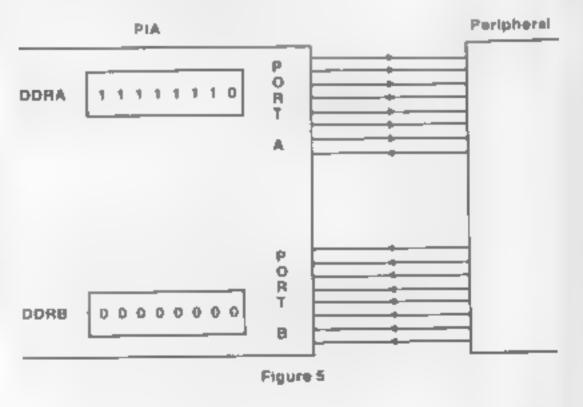
So we can see that the communications chip has a lot of work to do: converting data between serial and parallel forms, controlling data flow by means of handshaking lines, passing status information to the controlling computer and checking for errors.

Parallel I/O chips

Now we move from serial to parallel I/O. As with the previous section, there are a large number of chips to handle these functions. Probably the most common are the Motorola 6820 (Peripheral Interface Adaptor or PIA) and its equivalent, the MOS Technology 6520, and the 6522 Versatile Interface Adaptor, VIA.

There are almost as many uses for parallel I/O chips as there are functions in a computer. Internally, they are used for controlling lights, reading keyboards, turning on cassette motors, even sending data to and from the cassette itself. Externally, the Centronics standard parallel printer port is often controlled by a PIA or VIA as is the User Port on the Commodore range of computers.

Starting with the PIA, which is somewhat simpler, we find that the chip has 16 I/O lines, each of which can either be an input or an output. This gives added flexibility compared with the older Intel 8255, where the lines can only be programmed as inputs or outputs in groups of eight or four. If we want to use a VIA to control a piece of hardware which has seven inputs and nine outputs, we could program seven bits of port A as input, with the last bit together with the whole of port B set to output. This is done by writing values into two registers called Data Direction Registers.

out of paper, for example, or if a computer, it may be busy doing something else. Similarly, the transmitter may not have any data to send, which means that the receiver may waste a lot of time waiting for a message which will never arrive. To get round these problems a normal RS232 system has additional lines, called handshaking or control lines, which in effect say, "Yes I am ready to receive", or "No, I do not have any data for you".

Transmit Data	Receive Data
Receive Data	Transmit Data
Ground	Ground
Ready to send	Clear to receive
Clear to receive	Ready to send
Ready to receive	Clear to send
Clear to send	Ready to receive

Figure 4

All these lines go into the communications chip on the interface side of the system. From the processor side, the chip will look like a number of registers. There will be one or more control registers, a status register, a transmit register and a receive register. Let us follow the steps a computer has to go through to send a message to say, a printer. Firstly, it has to set up the chip according to the protocol expected by the printer. For example, different devices communicate at different speeds (baud rate) and have slightly different formats. The processor then has to tell the chip to raise the signal line which says "I have data to send". In the status register, one of the bits will correspond to the line from the printer saying whether it is ready to receive or not. The computer should test this bit to make sure that the printer is ready. There is another bit (transmitter ready) in the status register which tells the computer when the chip is ready for another byte of data. As far as the computer is concerned, the chip takes quite a long time (say 1-30ms) to send a character, the computer, which can send a character every 5 or 10 microseconds, has to spend a lot of time waiting around for the chip to finish sending the previous byte.

Figure 5

PIA — Peripheral

DDRA: 1 1 1 1 1 1 1 0 — PORT A

DDRB: 0 0 0 0 0 0 0 0 — PORT B

If we read the value in register A, we get ones and zeroes corresponding to the lines of port A. We say that these lines are level controlled. In addition to the 16 lines in ports A and B there are four control lines, CA1, CA2, CB1 and CB2. These inputs are called edge triggered inputs and they work slightly

different y If the CB1 input is set up to detect a positive transition, then the chip will set a bit in the control register B when the line goes from a 0 to a 1 However, even if the CB1 input goes low again the bit in the control register stays 1 until the processor specifically clears it. This is useful if the pulse we are looking for on the CB1 line only lasts for a short time If we tried to use a level input, the pulse might go past while the processor was doing something else so that we missed it completely In addition, CB2 and CA2 can also be programmed as outputs

A VIA contains all the features we have just described for the P A but with lots of new things as well! It has two 16-bit timers, which can be used for a wide range of tasks If we store a value into timer 1, then the chip will immediately start to decrement the value, and will signal the processor when it reaches zero Say we want to do something every 1/60th second, such as scanning the keyboard on our PET We set the timer to the required delay, and every time the timer reaches zero, the VIA will interrupt the processor which can then scan the keyboard, update the internal clock. sing "Rule Britannia" or whatever else we want

Instead of interrupting the processor, the VIA could give a signal on one of the output lines. In this way our computer can give an accurately defined series of pulses to a peripheral, without tying up the processor at all! We can also use one of the timers to count the number of pulses coming into the chip on a certain input again without wasting any processor time

The VIA also contains a "shift register" which allows the device to convert data between serial and parallel format It is not specifically designed for RS232 use as are the communications chips we discussed earlier but it can be used for serial communications within a computer system

Video Controllers
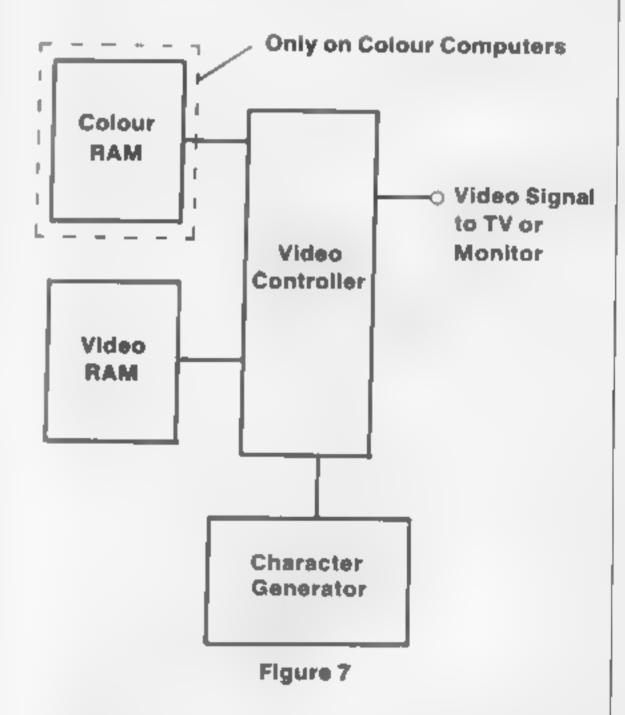
One thing every computer with a screen has in one form or other is a video controller It is the job of the video controller to turn an oddball collection of numbers inside the computer into a recognisable display on the screen. Video controllers, which are quite used to going about their business quietly and out of the public eye, have recently been very much in the spotlight with the introduction of the VIC (Video Interface Chip) which features prominently in the computer of the same name

The most important job of the video controller is actually forming characters on the screen A character is displayed as a pattern of dots in a square or rectangular matrix

Figure 6

Our "typical" computer uses an 8x8 matrix, because that will make things a little easier later on If we represent each light dot by a "1" bit, and each dark dot by a "0" bit, we can store the representation of the character as a group of 8 bytes. These representations, one for each possible symbol which can be displayed on the screen, are stored in a ROM inside the computer called the character generator ROM Now we know from our BASIC programming that each character has a corresponding code which we can find out using the ASC function, in the range 0 – 255 Say, for instance, we want to display the character "A" The code for "A" is 65, and each character occupies 8 bytes, so the representation to "A" starts 65 x 8 = 520 bytes into the character generator ROM

Here is a diagram of the video system of a typical computer

Figure 7
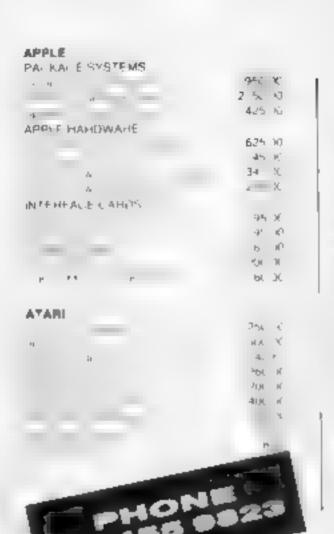
The video controller chip is continually scanning the video RAM area It looks at a location, picks up the character code and uses it to find the 8 byte character representation in the character generator It is able to use this to generate the correct pattern of light and dark dots to feed to the video output signal which eventually produces to picture on the screen. If the computer has colour facilities. then the video controller also has to pick up colour information from the colour RAM at the same time as it gets the character code from the video RAM. All in all, the video controller is quite a busy little chip!
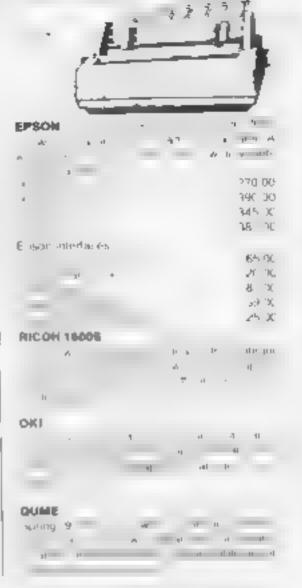
There are many other groups of interesting devices A/D and D/A converters, speech synthesis chips. modem chips and so on, but these tend to be specialist devices, which are really outside the scope of this article However, as these chips become more common in the personal computer market, we will keep you up to date on what they do and how they work

60

Business Briefing

Which Application?

Business Briefing is a new series for businessmen and professionals who 'think they need a micro' but don't know what to buy. In the first article **Charles Christian** looks at the most popular applications and how to make up your 'shopping list'.

If over the last few months, you have been reading this magazine as carefully as the Editor hopes, you must by now have grasped a reasonable idea as to what microcomputers are all about

No longer should you flush with embarrassment when the conversation turns to talk of "VDUs" "ROMs" and "RAMs" No more do you ponder why anyone in their right mind would want to buy a disk that has gone "floppy" Nor for that matter, do you believe any longer that the "silicon office" is just another name for the National Coal Board's headquarters Indeed if you have been particularly diligent you may have even reached the conclusion that a microcomputer could have a valuable role to play as a business machine in your own office.

But whilst buying computer hardware is an issue that seems to be discussed endlessly that is only half the problem For unless you are a budding "mastermind" and can write your own programs, you are also going to have to buy some software if your system is going to be of any use

And this is where the difficulties start, for in the UK today there are estimated to be over 1 000 different software houses and dealers selling literally hundreds of different types of program

Obviously a microcomputer is not something that needs to be confined to a corner of the wages department and only used once a week when Doris runs off the factory payroll On the other hand it is courting disaster to go out and immediately buy every program you think may be of use, as you will rapidly find yourself being submerged in a welter of paperwork as your computer spews out useless report after useless report

There is therefore a real need to strike a happy medium with the software you use, but just how do you set about separating the wood from the trees, or in some cases the grain from the chaff when it comes to selecting business programs? Naturally any final decision must always be determined by the particular needs of each individual company but it is possible to detect some general trends as to the way microcomputer users

select their first software packages, which could usefully be followed by others

Given the present state of the UK microcomputer market, there is still insufficient data to be able to categorically state which individual software program seems to sell better than others Consequently, I have grouped together business software programs into six basic categories These are, in alphabetical order

- database management
- financial planning
- general accounting
- professional applications
- specialist programs, and
- word processing

Before going on, for the benefit of the uninitiated, to a more detailed discussion of each of these categories in turn, I would just point out that I do appreciate that this classification omits what are probably the most widely used software programs in offices today Unfortunately, despite the popularity of PAC-MAN and BREAKOUT and others of that ilk on computer terminals up and down the length and breadth of the country, games packages do not qualify as genuine business software

Database Management

Despite its rather awesome sounding name a database management software system (you may also see it referred to as "information management"), is in fact a relatively straightforward concept

It can probably best be understood if you think of an electronic library or filing cabinet Thus, the software is structured in such a way that vast amounts of information can be recorded and stored away, with each item of information only needing to be recorded once This constitutes the database element To this is added the management element whereby the program contains an elaborate cross-referencing or indexing system. The end result is the creation of a pool of information to which the computer operator has ac-

cess and can draw upon in any one of a number of different permutations

For instance, one company with which am familiar is in the business of supplying fresh produce to some 1500 automatic vending machines in the Thames Valley and Home Counties area. On its database the company's computer records such information as the locations of all the machines, their distances from the company's service centre the type of produce they sell their best selling lines at each site and their relative profitability

When it comes to plotting a supply rota for the machines management consequently has a number of options open to them Machines could be supplied on a straightforward geographic basis, which would probably be the most obvious But it would be just as easy to draw up a rota where the crucial criterion was the size of the weekly takings or even the number of "Mars Bars" sold each day

Admittedly such an application may be rather too sophisticated for the average first-time computer user to begin with, but a database management system can still have a valuable role to play, even if it is going to be used for far more modest purposes For example, many companies, before they have got their accounts fully computerised use it supplied with information drawn from their books and ledgers to produce management reports such as listings by age of creditors and debtors

Financial planning

Financial planning (sometimes called "financial modelling", "what-if programs", "cash flow forecasting" and even the "electronic spreadsheet") is the generic term used to describe the increasing number of programs coming onto the market on the lines of *VisiCalc* and *MicroModeller*

Basically financial planning is a computer tool for calculating the effect changing certain variables would have on a business's future For example, what would happen if the price of petrol went up to £3 per gallon VAT increased by 5% or basic rate tax was cut by

5p in the pound?

By using financial planning software it is possible to follow hypothetical changes in these variables through the whole gamut of your budgetry process. For instance an increase in the rate of VAT will mean an increase in the sale price of a product, which will usually mean a decline in the volume of sales. This in turn will mean that cash income falls, stock levels rise, the overdraft grows, profits fall away and the directors don't get their bonuses at Christmas. Armed with this d small information you will at least be able to draw up contingency plans to minimise the impact of any such changes.

Obviously the effectiveness of such a program will very much depend upon the accuracy and the quality of the information you originally feed into the computer to act as the basis for your assumptions, and for this reason many computer users tend to employ financial planning as an adjunct to an established computerised accounting system.

General accounting

General or "integrated" accounting systems are, according to a recent survey, the most widely available of all software programs currently for sale on the UK microcomputer market. The reason for this being that almost every business organisation in the country is a potential purchaser of such a system.

Most accounting systems tend to be modular in construction, so that they can easily be adapted to meet the individual needs of different purchasers. The reason for this is that it is a complete waste of time having an elaborate "payroll" program in operation if your company only employs a handful of staff each of whom is paid a monthly salary by cheque. On the other hand if your business has several hundred employees on its books all of whom are paid weekly in cash and whose wages vary greatly depending on the number of regular and overtime hours they put in, then some form of automated payroll facility within the system would be a necessity. Calculations of tax and statutory deductions and special rates of pay, the completion of pay-slips and even the production of coin analyses to speed up pay packet preparation could then all be catered for.

For most businesses, the core on an accounting system will be a "bought" or "purchase" ledger program operating in conjunction with a "sales" ledger. These in turn could usefully be appended by "stock control", "order entry" and "invoicing" software so that cheque issuing, statements, sales analysis, credit control, VAT, cash flow and account posing work can all be processed speedily and with a minimum of effort. As already mentioned, it may also be appropriate to add payroll work to the list of tasks undertaken by computer.

Once such a system is in operation most companies usually take the process one stage further by drawing it all together under one general program which can automatically generate trial and nominal balance sheets and profit and loss accounts, as well as the other management reports companies need from time to time.

The beauty of such a system is that the data needed to generate such reports can automatically be drawn from the sales and purchase programs, hence the data only needs to be entered once. A factor which can considerably cut down the possibility of errors being made. An additional attraction is that the data produced by a general accounting system is an ideal source of raw material for database management and financial planning programs.

Professional applications

Whilst a general accounting system will meet the computing needs of most commercial and industrial concerns, there are still a number of individuals and businesses in this country, most notably members of the professions, whose work, or the formal regulations of their professional bodies, means that they are looking for something more from a computer.

Solicitors, for example, because they handle very large sums of money belonging to their clients, are compelled to comply with The Law Society's very stringent Solicitors Accounts Rules when they prepare their books and ledgers. With chartered and certified accountants, however, their problem is not so much the requirements of their profession as the practicalities of their work as, for example, one of their biggest headaches is the auditing of accounts for clients based upon "incomplete records".

To meet the needs of such people in recent years a thriving industry has grown up supplying off-the-peg software programs for professional applications. Currently one of the most popular of these is the "time-recording" system for the benefit of individuals and organisations that offer a service to a client rather than a product to a customer. Users include accountants, solicitors, architects and surveyors and a typical system would produce a record of hours spent and the amount of fees earned by different partners carrying out work for different clients.

Despite the fact that many of the professional applications cater for a relatively small section of the community, they tend to be proportionally more popular than more general software programs, precisely because the people using them are usually more conscious that as "time is money", non-rechargeable administrative work should be kept to a minimum.

Specialist programs

Although most first-time computer users will find that their software needs can be satisfied by widely available off-the-peg programs, there are some businesses with rather more special needs.

These in turn can effectively be sub-divided into three further categories -

users whose needs are catered for by very specialist software programs, such as "printers job costing" for example. The drawback with these, however, is that there may only be three or four suppliers handling it within the whole country, which can lead to maintenance and back-up problems.

– users whose needs are not quite met by off-the-peg programs and who may require some elements of "tailoring" to make the software compatible with the way they run their business. In practice this is a fairly common requirement.

– users whose needs are literally unique and who require "bespoke" software to be written specifically for them. The major difficulty with this option is that it can be prohibitively expensive.

Before selecting any of these options you must weigh up all the cost and inconvenience factors involved most carefully. Indeed some businesses find that it is easier in the long run to adapt their own administrative systems to run on a general purpose computer system rather than try to get a software

Making the best use of a micro involves choosing packages that will keep your micro constantly busy without congestion.

Which Application

program custom written to meet a more specific need

Word processing

Lastly but by no means least, there is the option of purchasing word processing (sometimes called "text editing" software In a nutshell, word processing is computer-enhanced typewriting

Instead of typing directly onto paper a text is entered onto a word processor and displayed on the VDU. Once there it can be edited, drafts can be printed off for approval and the text can be stored in the system's memory. Hours, days or weeks later the text can be retrieved and further amendments deletions, substitutions and re-arrangements of layout can be made before any number of top copies can then be run off for distribution

As such, word processing holds a considerable appeal for business and professional offices for instance those of solicitors, where long and complicated documents, such as contracts and conveyances may go through many stages of redrafting or contain many

Word processing is by far the most popular 'first' package

from a word processing system if it is combined with some form of indexed mailing list facility. A number of estate agents, for example, use such systems to selectively mail out lists of property to prospective purchasers. Thus those of us who can only afford rundown slum tenements in Tooting with hot and cold running cockroaches in each room are not plagued with estate agents particulars of luxurious office accommodation in Goring

Even if you are unlikely to be able to employ word processor software in such an elaborate way in your office, it should not automatically be ruled out as a possible option. Indeed, many business organisations have found that the investment in word processing facilities can be justified in terms of efficiency – no longer any danger of valuable drafts going missing, job satisfaction – no more secretaries tearing out their hair as they type out the same boring standard letters for the hundredth time, and general convenience even if they only have a fairly routine clerical work load

Purchasing priorities

So much for what the basic categories of

standard paragraphs

An even greater potential can be realised business software programs currently available can do But in what order should a First time buyer rank them on his shopping list?

As mentioned earlier the ultimate determining factor must be the particular needs of each individual customer However, to get some idea as to what many first-time buyers are currently doing, I contacted a number of different software companies and retail dealers around the country to find out what their experience was

In fact, despite the apparent abundance of outlets, carrying out the research proved to be quite a difficult task. Train strikes, annual holidays and heavy liquid lunches appear to have taken a heavy toll of the industry. Indeed, some offices seem to have become "the haunts of owl and bat" which is a smart slick way of saying no-one bothered to answer the phone

The league table

Despite the surfeit of accounting systems on the market, plus the fact that accounts work is one function that lends itself very easily to computerisation, the results of my survey show that general accounting software does not figure very highly on most people's initial shopping lists. In fact the outright winner by a clear margin turns out to be word processing packages

The full league table is as follows

1 Word processing, followed at some distance by
2 Database management, followed closely by
3 General accounting followed closely by
4 Financial planning, followed a long way behind by
5. Jointly - professional applications and specialist programs

The reason for this, perhaps surprising, result was explained by Steve Johnson, a director of Johnson Microcomputers, who are Camberley-based members of the new ICL Trader Point Scheme. Steve reckons that word processing and database management systems are popular "because it is possible to achieve almost immediate results with them"

This view was also expressed by Peter Wills of Mercator Computer Systems, a Bristol-based software house. Said Peter "Most people can become proficient at word processing after only a couple of days training but with accounts work, even if the system is easy to learn, it can still often take months to get it fully up and running Businesses nowadays tend to use computers for several different applications and recognise that as they have to start somewhere, it is best with a relatively simple application like word processing that is easy to get going "

Buying a computer is like getting married", Peter reckons, "it is best to iron out the wrinkles before you get hooked "

This whole phenomenon of starting with a simple application and then moving on to one more complex is probably best summed-up by the words of David Cater a director of Computaline of Ealing, West London, who handle the Sirius. He believes that one of the problems many first-time users face is that they are daunted by the prospect of working with a computer "but if you let a customer loose on word processing he quickly masters it and thereby overcomes the mystique of the computer Once that is done he is in a better frame of mind to start tackling accounts work and more specialist applications "
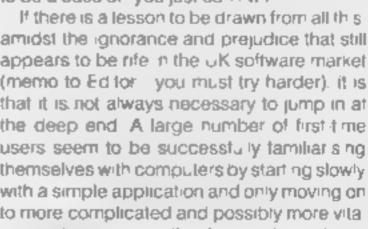
Key factor

One other point that was brought out during the course of my researches most notably by Peter Hartley, a consultant with Stage One Computers of Bournemouth and Jim Thompson of Commodore dealers The Microcomputer Centre of Sheen, South West London, was that a key factor in determining what programs software users select is "what stimulated them to come to the dealer in the first place"

The general concensus seems to be that about half of all customers have a specific idea about what they are looking for and have either read about a particular package in a magazine, had it recommended to them by a colleague who has a similar system, or have responded to an advertising campaign Whether the system is appropriate for them or not many of these customers are intent on buying one come what may

Quite a number of others tend to be people with a business problem but who have no real idea how computerisation can help them They only know that it can because "this is Information Technology Year and I heard about it on TV" Unfortunately anyone too vague about what they want seems to be equally difficult for some dealers to cater for. As one software consultant, who prefers to remain anonymous put it rather uncharitably "these people are a pain in the arse" Which seems to be a case of "you just can't win"

If there is a lesson to be drawn from all this amidst the ignorance and prejudice that still appears to be rife in the UK software market (memo to Editor you must try harder), it is that it is not always necessary to jump in at the deep end A large number of first time users seem to be successfully familiarising themselves with computers by starting slowly with a simple application and only moving on to more complicated and possibly more vital applications once the basics have been mastered

PLAYER MISS[...]

Every magazine writer bears a cross whatever he writes may well be out of date by the time it gets read. It's happened to me this time but not nastily.

By narrowing your eyes and looking very closely you may possibly have noticed that I've been extolling Atari's virtues for some time now. This was based on its qualities considerable) facilities (amazing) and price (reasonable)

It's the last of these that's risen to thump me for since my opening article in the series Atari's mysterious world without fear, series Atari have dropped their prices with what can cried by be called a crunch

The Atari 400 is now an astonishing £199.95 and that's with a full 16k of memory while the up market Atari 800 is down to £499 and Atari disk drives are less than £300

They're not likely to get lower and knowing what's inside and what you do I can't even see how they're managing these prices. And as I've been saying for months now it's what's inside that makes Atari such a creative computer

Which brings me neatly to a continuation of where we got to last month

Display lists continued

You'll remember we built a nice complicated display list last time which lets us use a number of different graphics modes all of one on the same screen

In doing that (and our example was necessarily complicated) you'd seldom want to know really how we got to the processes involved

Hopefully you found display lists were nowhere near as mysterious as you'd thought. You may even have been so devout venturous and played around to set up some different and simpler examples for your self

You may even have got bored just looking at blank sections of different modes on the screen and had a go at putting something in them

And at that point you may have run into some difficulty

It's a very well I hear you muttering get ting the screen divided into fun bars of dif ferent graphics modes but there's not much point unless we can actually put something useful there too

Getting stuff on the screen

Well you remember I said you could print or plot to a screen of mixed modes just so long as you didn't exceed the normal cursor range for the mode at which you were aiming

I went on to say you could find the vertical position by counting the mode lines from the screen top

The horizontal position you find in the usual way for the graphics mode you've targeted by starting from the left after checking how many horizontal lines across you want in that mode

That sounds nice and simple but what often as you may well have discovered the line on the screen at which you want to print on put just won't print at all or if it does it's called Murphy's law and other responsible things

So what's the answer then? Program Atari find its hard by just starting with the ory for the position you want and poking the information in. And that's what we're going look at now starting with player missile graphics

Where on the screen?

The first thing you have to do is work out the memory address from the top left hand screen corner

Put another way this means you have to find out where the heck in memory Atari's planned to stuff whatever you want into the screen position
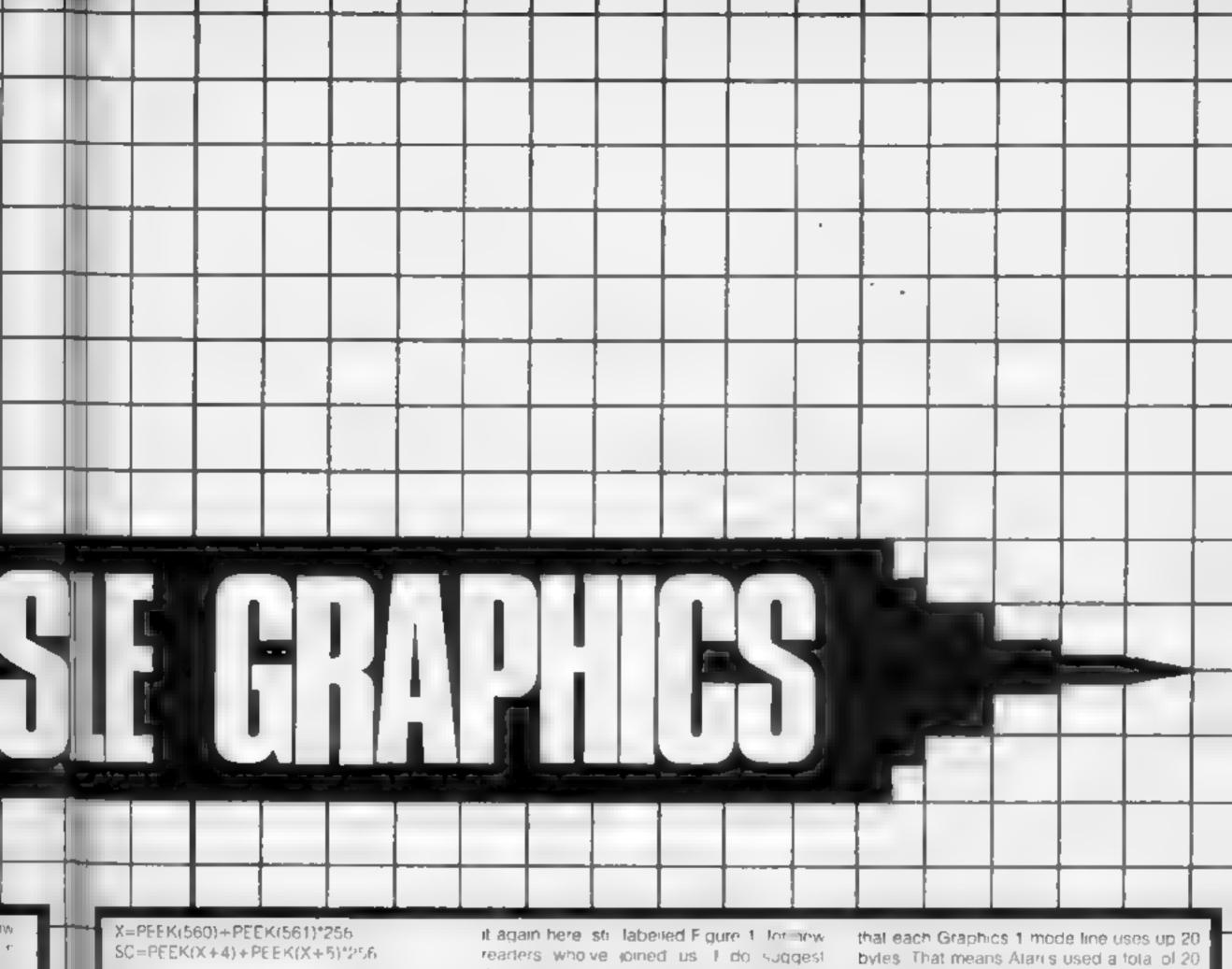
This would be a complicated display list you've put together but working it out is very very simple

If we call the memory address SC (for screen corner sophisticated stuff eh?) there are actually two ways to get the number you need but the second is almost always quickest

I'll give you both though because you never know there might come a time when you want a change

Here's the first and is given one way

X=PEEK(560)+PEEK(561)*256
SC=PEEK(X+4)+PEEK(X+5)*256

Now here's the one-line method of getting the same result

SC=PEEK(88)+PEEK(89)*256

Whichever method you use, you'll finish up with SC equal to the memory address for the top left-hand screen corner. Painless so far.

How much from here to there?

The next job is to work out how much memory will be used between the top left-hand screen corner and the place on the screen you'd quite like to have something appear.

Now this *does* appear a bit complicated at first think, because you'll pass through a number of different graphics modes before you get to where you want to be. Each of those modes will use up a different amount of memory, won't it?

Is this "oh dear" time? Not a bit of it.

Again there's nothing to it except you have to do just a tiny bit of counting. Not a lot, and here's how it's done.

We'll give a label to the place we want whatever it is to appear. Let's make it MA (for memory amount — more sophistication!) because it reminds us we're actually calculating the amount of memory before we get to our target position.

To do this working out you're going to need the table labelled Figure 1 in last month's *MicroComputer Printout*. We've reproduced it again here, still labelled Figure 1, for new readers who've joined us. I do suggest though that you get hold of the first article if you haven't got it. We'll be assuming from time to time that you're familiar with what it contained.

Right, using the information in Figure 1 you can now work out how much memory you've used in each of the different graphics mode lines in your screen display. Start with the top one and work your way down, adding the memory used by each separate mode line until you reach the actual line you want to put something on.

When you get to that line, stop adding memory figures together and count across the 20 odd way to your target position. Remember how many positions you could across, and add that figure to the total RAM you've used coming down from the top.

Let's have an example!

Now this all may have sounded a bit muddling. It's not a point since it's one of those things that's harder to explain than to do, and you won't have any problems.

Just in case though, let's take an example.

Suppose the whole screen was made up of Graphics 1 mode lines. It'd be pretty boring but it makes for a nice easy explanation. That's a...

You plan to put something on the second mode line down, in the third horizontal position. Ok so far? Fine.

Well there's 1 mode line above the one you want, and turning to last month's table we see

that each Graphics 1 mode line uses up 20 bytes. That means Atari's used a total of 20 bytes to the end of the mode line immediately above the one you're aiming a...

...used the 2 bytes to get across to the ... you want, which is a total of 22, you can see I went to a good school, can't I you?

Now we resuscitate our original calculation, the one that gave us a value for the position at the top left hand corner of the screen — and we add 22 to it.

Guess what? That's the screen memory value for the very position we want to put a character at. Great heavens, and you thought all this was hard! I told you it was easy.

The slightly harder bit comes now.

Take a poke at it!

I hope I haven't suddenly scared you because that was so easy. What we've done so far has been so easy that you've no need to fear this next bit.

You see, to get stuff on the screen we're going to have to poke it into screen memory. We now know *where* to poke it (if you'll pardon the expression). There remains only the question of what to poke.

We can't simply poke an "H" an "E" an "L" another "L" and an "O" to put "HELLO" on the screen. We have to poke numbers, and they're *not* the Atari code numbers you've seen in an Appendix to your Atari manual.

Instead we have to poke the number from an internal code set. Sorry, but that's the way it is.

►

GRAPHICS MODE NUMBER	SCAN LINES FOR EACH MODE LINE	MODE BYTE CODE NUMBER	RAM USED FOR EACH MODE LINE
0	8	2	40
1	8	6	20
2	16	7	20
3	8	8	10
4	4	9	10
5	4	10	20
6	2	11	20
7	2	13	40
8	1	15	40

FIGURE 1: Choose your graphics mode and this table tells you how many horizontal TV scan lines occur in each mode line; the special mode byte code number used in constructing display lists; and the RAM which each mode line uses.

ATARI ASCII (ATASCII) VALUE	ADDITION OR SUBTRACTION FACTOR
0- 31	Add 64 to value
32- 95	Subtract 32 from value
96-127	Use value
128-159	Add 64 to value
160-223	Subtract 32 from value
224-255	Use value

FIGURE 2: The conversion values to get screen poke figures from ATASCII codes.

```
10 DIM L$(1)
11 GRAPHICS 0
12 SC=PEEK(88)+PEEK(89)*256
13 POSITION 0,0
14 PRINT "VERTICAL POSITION ";
15 INPUT VP
16 POSITION 0,0
17 PRINT "HORIZONTAL POSITION ";
18 INPUT HP
19 POSITION 0,0
20 PRINT "WHICH LETTER TO POKE ";
21 INPUT L$
22 POKE SC+40*VP+HP,ASC(L$)-32
23 POSITION 0,0
24 PRINT "                    "
25 GOTO 13
```

FIGURE 3: A simple Basic program in Graphics 0 to poke characters to the screen. Change line 11 and experiment further in Graphics 1 and 2.

Never fear!

Fear not though - conversion from A is number and decimentors at the time on as you see if you take a quick look at Figure 2. It's a complete table for finding the equivalent internal code number from Atari's ATASCII numbers.

With a bit of ingenuity, you should even be able to program a few lines to do conversions for you!

The short program in Figure 3 doesn't do this it does a bit and assumes you're going to the shifted keys as I shows Thore will automatically far between the ATA and I your look give 2 again you see the set things of and it of ATASCII values give the right internal code for them.

Enter it and give it a try, but note that it's only in one graphics mode – Graphics 0. Then, when you've experimented enough put a display list together and have a go at poking characters direct to any of the other display mode lines you've used.

Remember that you can only use alphanumeric characters in certain graphics modes, but that's the whole point of doing this exercise. It lets you fill whatever part of the screen you choose with a high-resolution mix, and still have slices of screen for alphanumeric characters.

I could have given you a sample program to do it here, but be honest, it wouldn't have been as educational as doing it for yourself, would it? The whole idea is to get you familiar with display lists and what you can do with them before we move on to the slightly (but only slightly) more advanced stuff.

And here we go!

It's player-missile time, folks, but unhappily we simply have to get into some of that prefacing explanation again before we can go ahead with actually trying things for ourselves.

I'd like to avoid it but there's no way anything which follows will make real sense unless we look at some principles first.

And even then the approach will be very very simple – a full explanation of *everything*

PLAYER MISSILE GRAPHICS

involved in player-missile graphics would fill an issue or two of this magazine

If you really want to get deep into the subject - and quite a few people would - then I do suggest you get a copy of "De Re Atari"

Happily it's now on sale at many of the better computer shops, and there'll be few Atari purchases you'll come to value more. In the meantime you'll be able to get an idea of whether you want to go further by testing the water with these articles

First principles

Here's a question. Which is best. reading an article about player-missile graphics, or having someone actually show you? The answer's obvious, isn't it?

And that's why so-called player-missile graphics are themselves important. They give smooth fluid, demonstrative movement to a static computer screen probably filled with otherwise unmoving words and figures

Player-missile graphics aren't now used only in "shoot-em up" space games though that's where they originally got their name These days they give life to computer displays and make what's on the screen easier to understand

We can best see the principles employed by first checking what would happen in an ordinary computer if we wanted to move something around on the screen

Moving things around

The screen RAM would store an object on a background and display both. To move the object the program would repeatedly have to write the background into one side of the object and write the other side of the object into the background. This would result in movement a bit rather jerky

There's a significant problem though. A TV screen has two dimensions. The RAM controlling what's on it only has one

Remember those lines which make up the screen? They're drawn by a flying spot racing across from left to right and screen RAM is controlling whether the spot lights up (for the object) or doesn't (for the background)

Any object on the screen has height as well as width unless it's a single horizontal line and we won't bother with that!

If you think about it, this means the object memory is scattered through the background memory Manipulating all that isn't easy

Why not? Simply because the object memory is strewn through the background memory To move the object, the problem has to be continually calculate where its bits are and that takes time and spoils fluid motion

The answer: P-M graphics

Like all brilliant answers to thorny problems the player-missile solution to moving things about on the screen is spectacularly simple If screen RAM is one-dimensional you merely create a one-dimensional object

Pardon? I thought you said we weren't going to bother about horizontal lines?

Ah, we did. Though our object will appear to be two-dimensional on the screen it'll only be one-dimensional in screen RAM

It'll be in RAM in a table which is either 128 or 256 bytes long according to whether you want a shorter fatter object or a taller thinner one

Only part of the table will be the object the rest will be background. The whole table is put on the screen as a vertical band going from the top of the screen to the bottom

We can then move the object back and forth inside the table and also move the whole table around on the screen There is virtually no calculation of where the bits of our object are in screen RAM - we'll always know they're totally enclosed in the table!

And you can have four!

It's at this stage the Atari magic really starts to work, for you can have up to four objects on the screen simultaneously and move each and every one quite independently of the others. Try doing that on some other computers

Not enough? Alright, we'll add completely independent colour for each object, and makes those colours independent of the background colour over which they move How's that?

You want more? Well how about letting each object move in front of or behind any of the other objects according to your choice And while we're on the job we'll let any of them also move "behind" the background too if you want

What's that? You're not satisfied? OK we'll give each of the objects some "missiles" and let each object independently control its own missile We'll make the missiles different colours too, each matching up with its object regardless of where it goes on the screen Are you happy now?

You are? I should think so too for that panoply of possibilities is what the astonishingly simple idea of making the object one-dimensional in memory eventually produced And all those facilities are available in both Atari home computers. All you have to do is use them

As a last word for now...

Once again (to echo my words of last month)

Thanks for staying with me through this preamble It was necessary as you'll see next month, when we actually start programming some simple player-missile graphics for ourselves

In the meantime it would be too bad to simply leave you at this point without something to play with, so try the program in Figure 4

```
10 GRAPHICS 0
11 SETCOLOR 2,0,0
12 A=100
13 B=48
14 R=PEEK(106)-8
15 POKE 54279,R
16 P=256*R
17 POKE 559,46
18 POKE 53277,3
19 POKE 53248,100
20 FOR I=P+512 TO P+640
21 POKE I,0
22 NEXT I
23 FOR I=P+512+B TO P+518+B
24 READ X
25 POKE I,X
26 NEXT I
27 DATA 8,17,35,255,32,16,8
28 POKE 704,88
29 S=STICK(0)
30 IF S=15 THEN 29
31 IF S=11 THEN A=A-1:POKE 53248,A
32 IF S=7 THEN A=A+1:POKE 53248,A
33 IF S=13 THEN 38
34 FOR I=8 TO 0 STEP -1
35 POKE P+512+B+I,PEEK(P+511+B+I)
36 NEXT I
37 B=B+1
38 IF S<>14 THEN 29
39 FOR I=0 TO 8
40 POKE P+511+B+I,PEEK(P+512+B+I)
41 NEXT I
42 B=B-1
43 GOTO 29
```

FIGURE 4: A very simple Basic program to produce a single object (a "player") on screen, movable with a joystick. The cursor stays in view; note the object can move over it without erasing it! Movement is slow because the program is in Basic; machine code would be much faster.

Don't expect anything too spectacular and let me stress now that it isn't a game or you'll be writing to ask how to play it it's simple a very straightforward demonstration of how to go about setting up just one object that can be moved around with a joystick

Keep it to hand though, because next month we'll be looking at how it works and how you can do the same thing (or better) too!

until then, enjoy your computing

The West End
VIC 20
Centre

ENHANCING PET BASIC

Dave Wardill and **Dave Barrett**
continue their series — adding new
keywords to PET's BASIC. This month
they introduce two commands for use
in business applications, using a
single command word to replace a
whole series of subroutines.

The newcomer to file handling often has problems with planning and setting out his records. He finds that people's names are not all the same length, and he has problems knowing where to find items when he is inspecting the data he so carefully filed such a short time ago.

Wouldn't it be nice, he muses, if all the names he put in were the same length? Then he would always know where they end, and he could always pick up the next item in the record easily.

Then he probably tries to incorporate such a plan in his program, and promptly disappears under a welter of LEFT$ and RIGHT$ until his program becomes so cluttered that he loses heart.

After a good night's sleep, he then probably produces a subroutine which will pad out any name he inputs — only to find a new set of problems when he accidentally puts in a name which is too long, and which irrevocably displaces the next item in his record.

He might produce code as bad as this:

```
5000 INPUT NA$
5010 L=LEN(NA$)
5020 IF L<5 OR L>10 THEN ?"ERROR" GOTO
```

```
5030 NA$=NA$+"          "
5040 NA$=LEFT$(NA$ 10)
```

If all his strings are to have different maximum and minimum lengths, he will either have to write lots of little routines like this, or find out how to pass parameters in BASIC to a subroutine. (It's easy enough, if a bit clumsy.)

At this point, he is yearning either for the time and expertise to master this, or for a pair of BASIC words like LINEINP and PADINP. And here they are.

Example

Let's start with a look at LINEINP. Suppose that the record you are entering as data is always longer than 5 characters, but never longer than 10. LINEINP allows you to specify this and thereafter stops you from making any errors.

LINEINP "ENTER NAME", 10,5,NA$

The syntax is obvious from the example.

LINEINP "message prompt"; MAX, MIN, variable name

Once you have used this, you can't enter any string less than 5 or longer than 10 characters long.

Illegal Characters

Also, it is not possible to enter any eccentric characters as part of the string. Up and down cursor movements are not accepted, nor are the REVERSE or OFF keys, the HOME or CLR keys, the RUN or STOP keys, and the colon and quote.

Cursor left and right work as controls, however, and so do INSERT and DELETE.

Even length

At the same time, it could be very useful to make sure that all the names that are entered are all the same length. This is invaluable when using Relative (Random) files

The routine called PADINP takes care of this very nicely

        PADINP "ENTER NAME", 20,5,NA$

This will use much the same syntax as the earlier example. The maximum acceptable length is 20 characters, whilst the minimum is 5. The string is called NA$. However, when a legitimate string is entered, its length is brought up to the maximum automatically by the addition of a bland character which will not produce any error when it is recalled later. This is character 96, one of the unused PET characters in this context. It isn't ignored or cut off in the way that a normal 'space' is although it does not show up on the screen or on a printer

How to enter it

Once again, you can have this word as a part of the BASIC additions which we described in the June issue. If you want it as a machine code subroutine, to be entered as and when you need it then it can be typed in with the Monitor into locations $3000-. If you want to relocate it, you will either have to enter the Assembler code and assemble it in your desired location, or use the 'New locate' function on Extramon. If you are none the wiser, give us a ring on Durham 711380 and we may be able to send you a relocated copy

Listing

Once again, we have supplied the listing in a heavily annotated form. Just reading it through should be a help if you are beginning to tangle with machine code. We always found that we needed more documented programs to read when we began, so we hope that you find these useful. At the same time, why don't you look back at the earlier issues of *Micro-Computer Printout* which have the series on Machine Code in them? (Ed — July to December 81) It might be interesting to put the two together and work out what our program does don't you think?

Next month

Just for a change, we want to go back to our original theme of meddling with the chips in the PET. There is a way of recovering a part of the PET memory in ROM which can be used to store permanently any program which you regularly need. For instance, you could install DOS (UNIVERSAL WEDGE) in this space, or EXTRAMON if you do a lot of machine code writing. It would not use any of those precious spare sockets in the PET, nor would it take up any RAM. See you next month

```
2970         BNE NOTCOM
2980         LDA ESHFCOM    CHANGE A COMMA INTO A SHIFTED COMMA
2990 NOTCOM  CMP ESPACE
3000         BNE NOTSPC
3010 NOTSPC  CMP ESHFTSP    TURN A SPACE INTO A SHIFTED SPACE
3020
3030         BEQ ENDIN
3040         CMP ESHFRET    ;END?
3050         BEQ ENDIN
3060         CMP ELEFT
3070         BNE N
3080         JMP DOLEFT
3090 N       CMP ERIGHT
3100         BNE NO
3110         JMP DORIGHT
3120 NO      CMP EDELETE
3130         BNE N
3140         JMP DODEL
3150 N1      CMP EINSERT
3160         BNE N
3170         JMP DOINS
     N2      CPX MAXCHR     ;CAN WE INSERT
             BEQ UPAY
             BCC UPAY
             INX
     UPAY    STA BUFFER,X
             JSR PRINT
             INX
             JMP NULL
     ENDIN   LDX SCAN       ;AT END
```

Assembly Language listing for LINEINP and PADINP


```
4210         JMP NULL
4220 YES2    PLA
4230         STA TEMP
4240         DEC TEMP
4250         LDX MAXCHR
4260         DEX
4270 NEXT    LDA BUFFER,X
4280         INX
4290         STA BUFFER,X
4300         DEX
4310         DEX
4320         CPX TEMP
4330         BNE NEXT
4340         LDA EINSERT
4350         JSR PRINT
4360         LDA £
4370         STA NOINS
4380         INX
4390         LDA ESPACE
4400         STA BUFFER,X
4410         JMP NULL
4420 SCAN    LDY £
4430 NEXT    LDA BUFFER,Y
4440         CMP EPADCHR
4450         BEQ GI
4460         INY
             CPY MAXCHR
             BNE NEXT
             BCC NEXT
     DOIT    CPY MINCHR
             RTS
     NORMOD  BYTE UP DOWN LEFT ON DELETE HOME    R,RVS OFFRVS
             BYTE
     PRNT    JSR PRINT
             LDA EDEL       ;SEE
             JSR PRINT
             JMP PRTPRT
     END
```

Machine Code Hex dump

In 1964, IBM introduced Series 360, so called on the theory that any customer could obtain a computer of the right size for his problems, and when his load went up switch to a more powerful system in the series without any change in software. The system was said to be compatible

This was a powerful and new idea, hence the 360 as in 360 degrees

Years passed and IBM introduced the 370 series, which cynics in the company promptly dubbed as covering all directions plus ten degrees

Naming computers can be fraught with problems. The reason more usually has to do with the fact that the computer business is international, and that a carefully selected name which has one connotation in the original manufacturer's language may have other connotations in other languages

Nobody has been as badly caught over this as Fujitsu were to be when they initially announced their FACOM range of computers. The name you might think is suitably innocuous. Not so. It all depends on the pronunciation

At announcement time the spokesman pronounced FACOM to rhyme with F*** em. A fluent English speaking Japanese present pointed out that this pronunciation had unintended overtones

There was hurried consultation, and Mr Spokesman got up again. This time he talked of the Fake em range

Having recovered from the shock the same Japanese gent y pointed out that this would not do either. Nowadays Fujitsu employees when addressing the English market tend either to talk about our range of computers without naming them, or else speak very carefully and clearly enunciate the name. Fa as in fallacy and com as in competition

All that this proves, of course, is that words can be an endless source of confusion, and no more so than in computing, for as every reader will know the computer is a literal beast. Talk to it about elbow bending and it will think that the subject under discussion is bending elbows

Translation by computer has been a goal of the computer community for over a quarter of a century and much money and effort has been expended. And naturally mistakes have abounded. Most of the stories are apocrypha, among them that of the computer which faced with the Russian words for hydraulic ram translated them as watery sheep. And then there was the system which translated "The spirit is willing, but the flesh is weak" as "The wine is agreeable, but the meat is lousy"

However, in computer translation, as in most things, truth is as ever stranger than fiction. Much of the early computer translation effort had its origins in the US defence programme and the need to find a rapid way of translating Russian literature and periodicals rapidly

In the early sixties, the Rome Air Defence Centre of the US Air Force came across the name Ava Gardner in a Russian paper. It looked for its nearest Russian equivalent, being unable to find an exact match in its English thesaurus, and came up with Ava Burning Bottom

And while we are on the subject of language did you ever see the cartoon (I have lost the reference so I cannot properly attribute this is) of the two US Generals down in the nuclear war centre at Colorado Springs

Outside, a nuclear explosion is going off inside the two generals are poring over a print out and one is saying to the other "That's not what it says here"

And while we are on the subject of words it's back to the Bible time

There is an appropriate biblical tag for almost anything that's common parlance in computing. You don't believe me? Then how about the following

Project proposal: Your old men shall dream dreams, your young men shall see visions Joel 2 28

Feasability study: Yet what I shall choose I wot not, for I am straight betwixt two Philippians 1 22,23

System specification: For which of you sitteth not down first and counteth the cost whether he have sufficient to finish it? St Luke 14 28

System audit: Surely thou has greatly deceived this people Acts 7 41

System design: And the rain descended, and the floods came and the winds blew and beat upon that house and it fell not for it was founded on a rock St Matthew 7 25

Implementation. And there shall be a time of trouble such as never was Daniel 12 1

As for the staff's Christmas bonus — you should be so lucky —

Then I looked on all the works that my hand had wrought and on the labour that I had laboured to do and there was no profit under the sun Ecclesiastes 2 11

Where to buy your CBM-PET

READ/WRITE

◁ 21

ex sts only in the Marketing Manager's imag
nation. as seems to be the case with the BBC
Micro

Ca um Steen
Helensburgh

*Regrettably this is only one of the many com
plaints we have received regarding the long
delays on the BBC Model B – and undoub
tedly there is little we can do about it, except
perhaps publish Mr Steen s letter as a warn
ing to potential buyers*

*We did pass on a copy to Acorn, in the
hope that they could clarify the situation. The
following reply comes from John Coll
Manager – Educational Services" and we
leave readers to draw their own conclusions*

"We share the frustration expressed by Mr
Steen. Model A s are currently produced by
Cleartone and to date about 10 000 have
been shipped and we are fully up to date with
orders

Model B s are unfortunately another story
ICL are responsible for their manufacture
and we have been quite unable to encourage
them to ship the agreed volume. In practice
we have received about 700 per week
though a figure above this was achieved
once during April We have had repeated as
surances that production would be in
creased but have come to realise that it is
necessary to look elsewhere for extra volume
in view of the enormous number of orders we
hold. We have now contracted RACE Elec
tronics as an additional UK source Cleartone
have been taken over by AB Electronics and
it is clear that AB intend to increase
throughput at the first opportunity

Nothing would please us more than to be
able to deliver computers in adequate vol
ume and actually to start promoting the
machine in the UK Mr Steen will be aware
that we have not advertised the machine
since last September We very much hope
that we will be able to achieve adequate vol
ume from UK sources "

The Editor welcomes your letters,
but if you require a personal reply
please enclose an S.A.E.

TOMMY'S TIPS

◁ 46

the record if you have a so used the B parameter), the DOS
transfers the data to the disk Now one reason why the DOS
cannot pick up the record length from the length of the
string you send to it is that you do not have to write a
complete record! The purpose of the B parameter is to
allow you to position the DOS pointer to the middle of a
record For example, the record might have a field in the
middle which holds a date of some sort. and that is all you
want to modify Now you do not want to waste time reading
the whole record into BASIC, modifying the field us ng
LEFT$ and MID$, and rewriting the whole record, when al
you have to do is write the new field to the middle of a
record using the B parameter

What happens when things go wrong? Well if the
record length you specif ed in the OPEN command was not
the same as that used when the file was created, all k nds of
things can happen DOS will use the length you gave it to
work out where the record it wants starts but this wil
probably be in the middle of a record on the d sk! The
problem now lies in the INPUT statement which will give up
when it finds a carriage return character What will happen
is that the INPUT statement will only read *part* of the record
from where DOS thinks the record starts up to the next
carriage return, so that you end up with a short record On
the other hand if you are writing to the disk all the data n
your print statement will be transferred, so that you will
overwrite the start of the next record on d sk

HOTLINE

◁ 28

SINCLAIR REGISTER

Pity the poor Editor Apart from
the distressing state of his biceps
and his shoes (he's working on the
former, we have started a fund for
the latter), he has to answer all
your letters

So we'll do him a favour by
plugging the ZX80/81 Register a
classified list of programs,
hardware sources, ZX publications
and User Clubs for the Sinclair

system It should answer al
reasonable questions on the
subject

The *Register* costs £2 95 from
its improbably named publishers
British Heritage Ph latelics, who
reside, with their large stamp
collection, at 2 Woodland Way,
Gosfield Halstead Essex

I will keep you posted on the
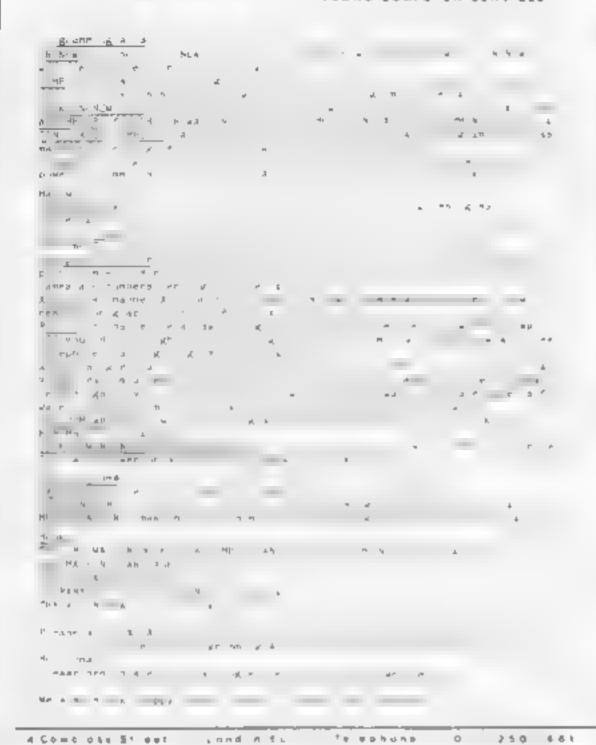Editorial biceptual situation

HAVE YOU MOVED?

If you are moving house,
please be sure to let us
know your new address so
that your copies of
MicroComputer Printout
can be redirected. It would
be helpful if you could en-
close your pre-
vious wrapper.

Don't miss out.

SUBSCRIBE!

FA HONOURS

For those suffering from severe soccer withdrawal symptoms as a result of the World Cup being over and the new season not yet arrived, this program may go some way to alleviate the distress. It is a small database of the various glories that soccer clubs have achieved and will allow you to wallow in nostalgia, look forward in hope or simply win a pint over a bet.

Data for seasons stretching from 1889 to 1982 is held covering the FA Cup finalists League Cup finalists Division 1 League Champions, and the highest position reached by each club currently in the four divisions. Regretfully lack of space and time has precluded me from including Scottish teams but it is a matter that you can easily attend to, given the information contained in this article

Team information can be obtained for each or all of the above 'honours' and listed by team or season. The data is held in these formats

**FA Cup and
League Cup:** YYSSRR where YY is the 2digit year number (e.g. 81 indicates 1980-81 and so on). SS is the score digit 1 is the team's score, digit 2 is the opponent's score. RR is the pointer to the rival team's data

Div. 1 winner: YY where YY is the year number as above

Highest position: YYDPP where YY is the year number as above. D is the Division (5 is old Div 3 North 6 is old Div 3 South). PP is the position reached. Where a club has reached this position more than once, only the most recent is recorded

By Bob Chappell

The data for the cups and championship is held as continuous strings of 6 digit multiples (there is only a single entry for highest position data). Thus for each team, the data is held as follows

Team name, FA cup data, League cup data, Div 1 data, highest position data

Where there has been no achievement in a category by a particular club, the data is signified by a single asterisk. The program does all the unpacking and places the data into the correct array elements

All the data is read in at the beginning of the

program and the name of the team currently being processed is flashed on the screen as an indication that the program is setting up the information. At this time, the program uses an inverted list technique. In order to access speedily the data by year as well as team and category, a mechanism of determining which teams were involved in a particular year is needed. This could be accomplished by simply searching through every team's data for the year in question but this would be rather slow. Instead the team data is searched at the beginning of the program and for every year item encountered an entry is made in a year array which points back to that team. Thus, item 2 in the year array is for 1901 (elements 1-83 hold years 1900-1982 and elements 90-100 hold years 1889-1899 respectively.

In each year element a string is built up of pointers back to the appropriate team data. For example, the data in year element 1 contains 031674 these are pointers to teams 03 (Aston Villa) 16 (Bury) and 74 (Southampton) So instead of having to search through every team for an event in the season 1899-1900 we can just examine this element in the year array which in turn leaves us to look only at Aston Villa Bury and Southampton

If there happens to be no entry in the year array or no corresponding entry for the category we are searching, then this means that that event was not held that year (in the case of the League Cup the message is displayed that it had not been instituted at that time

When searching by team, there is no need to enter the full name of the team, just enough characters to make it unique For example if you wish to display Liverpool's achievements, entering LIV is sufficient However

you must be careful — entering LI only will match against LINCOLN and the program will display their results as they are stored higher in the array than Liverpool. Similarly, L will produce the results for Leeds. Examine the list of names in the program, particularly those for Queen's Park Rangers and Nottingham Forest.

As far as I know, all the data is correct — if you have any difficulties, please first check that the data has been typed in correctly. If you disagree with any of the facts, please write to me care of the Editor.

(Ed — MicroComputer Printout does not undertake responsibility to reimburse expenses, medical or otherwise, resulting from errors in this program!)

For those of our readers not suffering severe soccer withdrawal symptoms, the program listing and documentation should at least provide an insight into some interesting techniques for handling lists of data.

Major Routines

20-75	Menu of options
80-145	Select by year — line 135 is start of year loop
150-210	Display FA Cup results by year
215-285	" League Cup results by year
290-345	" Championships by year
350-405	" Highest Position — line 405 is end of year loop
410-455	Subroutine to convert 1 into 1st, 2 into 2nd, etc
460-515	Obtain & search for team name
520-575	Display FA Cup results by team
580-600	" League Cup results by team
605-635	" Championships by team
640-655	" Highest position by team
660-730	Read in data and create inverted year list
740-1255	Team data
1260	Delay routine
1265-1275	Pause routine

Major Variables

T$(92,5)	Team array — for each of 92 teams contains FA Cup, League Cup, Div. 1 winner and highest position data.
TY$(100)	Year array containing pointers to team array
SY & SY$	Start of year range requested
TN$	Team name extracted
HS & AS	"Home" team and rival team's score
RT	Pointer to rival team
TN	Pointer to "home" team
N	Number of option selected

Reverse heart is Clear screen
Reverse R is reverse on
Reverse S is Home cursor

Program occupies about 11.5K, and was originally written to run on a PET, but can easily be converted.

```
5 rem ***football honours***
10 rem ***bob chappell 28/5/82***
15 goto665
20 rem **menu**
25 printhd$:print:print:print:print" 1. fa cup final results."
30 print:print" 2. league cup final results."
35 print:print" 3. league champions.":print:print" 4. highest position."
40 print:print" 5. all the above."
45 print:print:input"which number";n$:n=int(val(n$))
50 ifn<1orn>5thenprint:print"invalid entry.":gosub1260:goto25
55 print"C access:-":print:print:print:print" 1. by year."
60 print:print" 2. by team."
65 print:print:input"which number";p$:p=int(val(p$))
70 ifp<1orp>2thenprint:print"invalid entry.":gosub1260:goto55
75 print"C":print:print:ifp=2goto465
80 rem ***obtain year range***
85 print"starting year (enter 4 digits).":print
90 input"season ended in year";sy$:sy=int(val(sy$))
95 ifsy<1889orsy>1982thenprint:print"not in my records.":gosub1260:goto75
100 print:print"finishing year (enter 4 digits).":print
105 input"season ended in year";ey$:ey=int(val(ey$))
110 ifey<1889orey>1982thenprint:print"not in my records.":gosub1260:goto75
115 ifsy$>ey$thenprint:print"invalid entry.":gosub1260:goto75
120 rem ***fa cup finals by year***
125 sy=val(right$(sy$,2)):ey=val(right$(ey$,2))
130 ty=val(ey$)-val(sy$)+1
135 y=val(sy$)-2:j=sy:forj4=1toty:y=y+1:j=j+1:ifj>100thenj=j-100
140 tn$=ty$(j):y1=y:y=y+1:y$="R"+str$(y)+" -"+str$(y1)+" ":print"C";y$
145 onngoto150,225,300,360,150
150 n=0:print:print"R fa cup final ":print:iftn$=""goto210
155 forj1=1tolen(tn$)step2
160 tn=val(mid$(tn$,j1,2)):t$=t$(tn,1):o$=t$(tn,2)
165 ifo$=""goto205
170 forj2=1tolen(o$)step6:t1=val(mid$(o$,j2,2))
175 ift1<>j-1goto200
180 ns=val(mid$(o$,j2+2,1)):as=val(mid$(o$,j2+3,1))
185 ifns<asgoto200
190 rt=val(mid$(o$,j2+4,2))
195 as=1:printt$;hs;t$(rt,1);as;:j2=len(o$):j1=len(tn$)
200 nextj2
205 nextj1
210 ifn=0thenprint"no fa cup this season."
215 rem ***league cup finals by year***
220 ifn=1goto405
225 n=0:print:print"R league cup final ":print:iftn$=""goto285
230 forj1=1tolen(tn$)step2
235 tn=val(mid$(tn$,j1,2)):t$=t$(tn,1):o$=t$(tn,3)
240 ifo$=""goto280
245 forj2=1tolen(o$)step6:t1=val(mid$(o$,j2,2))
250 ift1<>j-1goto275
255 hs=val(mid$(o$,j2+2,1)):as=val(mid$(o$,j2+3,1))
260 ifns<asgoto275
265 rt=val(mid$(o$,j2+4,2))
270 as=1:printt$;hs;t$(rt,1);as;:j2=len(o$):j1=len(tn$)
275 nextj2
280 nextj1
285 ifn=0thenprint"the league cup had not been instituted."
290 rem ***div 1 champions by year***
295 ifn=2goto405
300 n=0:print:print"R league champions ":print:iftn$=""goto345
305 forj1=1tolen(tn$)step2
310 tn=val(mid$(tn$,j1,2)):t$=t$(tn,1):o$=t$(tn,4)
315 ifo$=""goto340
320 forj2=1tolen(o$)step2:t1=val(mid$(o$,j2,2))
325 ift1<>j-1goto335
330 z=1:printt$;:j2=len(o$):j1=len(tn$)
335 nextj2
340 nextj1
345 ifz=0thenprint"no championships this season."
350 rem ***best position***
355 ifn=3goto405
360 z=0:print:print"R best position ever ":print:iftn$=""goto400
365 l$="":forj1=1tolen(tn$)step2
370 tn=val(mid$(tn$,j1,2)):t$=t$(tn,1):o$=t$(tn,5)
375 t1=val(mid$(o$,1,2)):ift1<>j-1goto395
380 ift1=1goto395
385 z=1:l$=t$:printt$;" finished ";:gosub415
390 printp$;" in div.";d$
395 nextj1
400 ifz=0thenprint"nothing for this season."
405 gosub1265:nextj4:goto25
410 rem ***position subroutine***
415 p$=mid$(o$,4,2):ifp$<"10"thenp$=right$(p$,1)
420 d$=mid$(o$,3,1):d=val(d$)
425 ifright$(p$,1)="1"andp$<>"11"thenp$=p$+"st":goto445
430 ifright$(p$,1)="2"andp$<>"12"thenp$=p$+"nd":goto445
435 ifright$(p$,1)="3"andp$<>"13"thenp$=p$+"rd":goto445
440 p$=p$+"th"
445 ifd=5thend$="n"
450 ifd=6thend$="a"
455 return
460 rem ***obtain team name***
465 print:print"type * to obtain all teams.":print:print:zz=0
470 input"the team is";a$:l=len(a$)
475 print:print:ifa$="*"thenforj3=1tott:zz=j3:gosub495:nextj3:goto25
480 forj=1tott:ifa$=left$(t$(j,1),l)thenzz=j:j=tt
485 next:ifzz=0thenprint"i have no record for that team.":gosub1260:goto25
490 gosub495:goto25
495 print"C";:ifn=1orn=5thengosub525
500 ifn=2orn=5thengosub585
505 ifn=3orn=5thengosub610
510 ifn=4orn=5thengosub645
515 gosub1265:return
520 rem ***fa cup finals by team***
525 print"R fa cup final - ";t$(zz,1);" ":a$=t$(zz,2)
530 ifa$=""thenprint:print"they have never reached the final.":goto540
535 gosub545
540 print:return
545 print:forj=1tolen(a$)step6:y=val(mid$(a$,j,2))+1900
550 ify>1983theny=y-100
555 hs=val(mid$(a$,j+2,1)):as=val(mid$(a$,j+3,1)):rt=val(mid$(a$,j+4,2))
560 as=val(mid$(a$,j+3,1)):rt=val(mid$(a$,j+4,2))
```

```
565 printy-1;"-";y;:ifha>asthenprint"beat ";:goto575
570 print"lost to ";
575 printt$(rt,1);ha:next:return
580 rem ***league cup finals by team***
585 print"R league cup final - ";t$(zz,1);" ";a$=t$(zz,3)
590 ifa$=""thenprint:print"they have never reached the final.":goto600
595 gosub545
600 print:return
605 rem ***league champions by team***
610 print"R league champions - ";t$(zz,1);" ":print:a$=t$(zz,4):z=0
615 ifa$=""thenprint"they have never won it.":return
620 l=len(a$):forj=1tolstep2:y=val(mid$(a$,j,2))+1900:z=z+1
625 ify>1983theny=y-100
630 printy-1;"=";y;:ifz>3thenz=0:print
635 next:print:return
640 rem ***best position by team***
645 print:print"R best position ever - ";t$(zz,1);" ":print:c$=t$(zz,5)
650 gosub415:y=val(mid$(c$,1,2))+1900:ify>1983theny=y-100
655 printy-1;"-";y;" finished ";p$;" in div.";d$:return
660 rem**initialise**
665 hd$="O R football honours   1889 - 1982 ":printhd$
670 print:print:print:print"please wait while i set up the data.
675 ty=100:tt=92:dimt$(tt,5),ty$(ty)
680 forj=1tott:readt$(j,1),t$(j,2),t$(j,3),t$(j,4),t$(j,5):next
685 forj=1tott:print"C":print:print:printt$(j,1)
690 j$=mid$(str$(j),2):iflen(j$)<2then j$="0"+j$
695 forj1=2to3:c$=t$(j,j1):ifc$=""goto710
700 forj2=1tolen(c$)step6:a=val(mid$(c$,j2,2))+1
705 ty$(a)=ty$(a)+j$:next j2
710 next j1:c$=t$(j,4):ifc$=""goto725
715 forj=1tolen(c$)step2:a=val(mid$(c$,j1,2))+1
720 ty$(a)=ty$(a)+j$:next6,j1
725 a=val(lef t$(t$(j,5),2))+1
730 ty$(a)=ty$(a)+j$:next:j:goto725
735 rem**team data**
740 dataaldershot,*,*,*,74308
745 dataarsenal
750 data270118300039521252361071150204552015271214572014270147193248800187
755 data680142691380,31333435384853 71,71101
760 dataaston villa,9203869510869732310520521310782010392402572148
765 data6132696313057102527510557732 35,94969799001081,81101
770 databarnsley,100252122086,*,*,15203
775 databirmingham,311286561347,633103,*,56106
780 datablackburn,906172913157283139600390,*,1214,14101
785 datablackpool,482448510252534308,*,*,51103
790 databolton,941457040147232087261047292063533407582048,*,*,21103
795 databournemouth,*,*,*,72303
800 databradford,111052,*,*,11105
805 databrentford,*,*,*,36105
810 databrighton,*,*,*,82113
815 databristol city,090148,*,*,07102
820 databristol rovers,*,*,*,59206
825 databurnley,1410454701206213 82,*,2160,60101
830 databury,004074036029,*,*,26104
835 datacambridge,*,*,*,80208
840 datacardiff,250171271002,*,*,24102
845 datacarlisle,*,*,*,75122
850 datacharlton,461429471015,*,*,37102
855 datachelsea,15037167128270 2142,653243721277,55,55101
860 datachester,*,*,*,36502
865 datachesterfield,*,*,*,47204
870 datacolchester,*,*,*,57603
875 datacoventry,*,*,*,78107
880 datacrewe,*,*,*,93210
885 datacrystal palace,*,*,*,80113
890 datadarlington,*,*,*,26215
895 dataderby,98135699147103061646 4120,*,*,7275,75101
900 datadoncaster,*,*,*,02207
```

```
905 dataeverton,93019097230306105207127233304766 3272660186
910 data772303,9115283239637 0,70101
915 dataexeter,*,*,*,33602
920 datafulham,750287,*,*,50110
925 datagillingham,*,*,*,75304
930 datagrimsby,*,*,*,35105
935 datahalifax,*,*,*,71303
940 datahartlepool,*,*,*,57502
945 datahereford,*,*,*,77222
950 datahuddersfield,20010322106528430630020238 0165,*,242526,26101
955 datahull,*,*,*,10203
960 dataipswich,781002,*,62,62101
965 dataleeds,651245701221721002730178,681002,6974,74101
970 dataleicester,4913906102826313 48690147,644377652321,*,29102
975 datalincoln,*,*,*,02205
980 dataliverpool,14011550020265214271120274305277124 8,780156812187823182
985 data010622234764667 33677798082,82101
990 dataluton,591256,*,*,58108
995 datamanchester city,0410082601083303313421635513 5256310569104 3812382
1000 data702186741290762152,3768,68101
1005 datamanchester utd,09101348420757120358020863314 3760174772145792302,*
1010 data08115256576567,67101
1015 datamansfield,*,*,*,78222
1020 datamiddlesbro,*,*,*,39104
1025 datamillwall,*,*,*,72203
1030 datanewcastle
1035 data0502030601310813901020041101102420033221025 12007521002553147740345
1040 data761247,05070927,27101
1045 datanewport,*,*,*,47222
1050 datanorthampton,*,*,*,66121
1055 datanorwich,*,62406873018275010 3,*,80112
1060 datanotts forest,983129592146,781045793274800190,78,78101
1065 datanotts county,911306944108,*,*,91103
1070 dataoldham,*,*,*,15102
1075 dataorient,*,*,*,63122
1080 dataoxford,*,*,*,75208
1085 datapeterborough,*,*,*,78304
1090 dataplymouth,*,*,*,53204
1095 dataportsmouth,29020834124739 4190,*,4950,50101
1100 dataport vale,*,*,*,31205
1105 datapreston,8930902201393713783810395423866 42387,*,8990,90101
1110 dataqpr,820182,673286,*,76102
1115 datareading,*,*,*,27214
1120 datarochdale,*,620455,*,27502
1125 datarotherham,*,612303,*,55203
1130 datascunthorpe,*,*,*,62204
1135 datasheffield utd,994129011382022174153021251018360102,*,98,98101
1140 datasheffield wed,9016069621900721313542866 62331,*,03042930,30101
1145 datashrewsbury,*,*,*,80213
1150 datasouthampton,000416021271761048,792356,*,71107
1155 datasouthend,*,*,*,50603
1160 datastockport,*,*,*,06210
1165 datastoke,*,643443722121,*,47104
1170 datasunderland,130103373165731042,*,929395021336,36101
1175 dataswansea,*,*,*,82106
1180 dataswindon,*,693102,*,70205
1185 datatorquay,*,*,*,68304
1190 datatottenham,013171211109061204362311567212181324782 1066
1195 data7120037310558213 45,5164,61101
1200 datatranmere,*,*,*,39222
1205 datawalsall,*,*,*,99206
1210 datawatford,*,*,*,82202
1215 datawest brom,9230039501031202041121053524725432 65681031
1220 data665987672356701247,20,20101
1225 datawest ham,2302086432657520338 01002,663586811245,*,73106
1230 datawigan,*,*,*,82803
1235 datawimbledon,*,*,*,82321
1240 datawolves,890365931031961272083152210182391463493143603006,742147801056
1245 data545859,59101
1250 datawrexham,*,*,*,79215
1255 datayork,*,*,*,75215
1260 forj=1to2000:next:return
1266 print:print"      press space to continueH"
1270 getb$:ifb$<>" "then1270
1275 return
```